NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

OPTIMIZATION AND SENSITVITY ANALYSIS FOR A LAUNCH TRAJECTORY

by

Thomas C. Manemeit

December 2014

Thesis Co-Advisors:　　　　　　　　　　Mark Karpenko
　　　　　　　　　　　　　　　　　　　　I. Michael Ross

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE		*Form Approved OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.		

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE December 2014	3. REPORT TYPE AND DATES COVERED Master's Thesis
4. TITLE AND SUBTITLE OPTIMIZATION AND SENSITVITY ANALYSIS FOR A LAUNCH TRAJECTORY		**5. FUNDING NUMBERS**
6. AUTHOR(S) Thomas C. Manemeit		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000		**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A		**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB protocol number ____N/A____.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	12b. DISTRIBUTION CODE A

13. ABSTRACT (maximum 200 words)

Using modern algorithms, an ideal launch vehicle trajectory can be calculated based on the principles of optimal control theory. Conventional approaches, such as shooting, seek to find the solution to a Hamiltonian boundary value problem. Finding solutions to a boundary value problem can be time consuming and difficult due to the twin curses of sensitivity and dimensionality. In an effort to alleviate these problems, pseduospectral optimal control theory can be used to reduce the time and effort required to design optimal launch trajectories. Problem formulation is shown to be a key step in this process. To illustrate the idea, a launch vehicle trajectory optimization problem is solved for maximizing the final velocity of the first stage of a multi-stage rocket assuming that all fuel will be expended. The sensitivity of the solution to uncertainties is examined by modeling environmental uncertainties as Gaussian processes in a Monte Carlo simulation. Combining optimal control and Monte Carlo analysis improves the planning process by allowing for worst case scenarios to be identified and mitigated.

14. SUBJECT TERMS astrodynamic optimization, launch vehicle, trajectory generation, DIDO			**15. NUMBER OF PAGES** 73
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

i

THIS PAGE INTENTIONALLY LEFT BLANK

OPTIMIZATION AND SENSITVITY ANALYSIS FOR A LAUNCH TRAJECTORY

Thomas C. Manemeit
Lieutenant, United States Navy
B.S., Saint Louis University, 2007

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ASTRONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 2014

Author: Thomas C. Manemeit

Approved by: Mark Karpenko
                 Thesis Co-Advisor

                 I. Michael Ross
                 Thesis Co-Advisor

                 Garth V. Hobson
                 Chair, Department of Mechanical and Aerospace Engineering

iii

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Using modern algorithms, an ideal launch vehicle trajectory can be calculated based on the principles of optimal control theory. Conventional approaches, such as shooting, seek to find the solution to a Hamiltonian boundary value problem. Finding solutions to a boundary value problem can be time consuming and difficult due to the twin curses of sensitivity and dimensionality. In an effort to alleviate these problems, pseduospectral optimal control theory can be used to reduce the time and effort required to design optimal launch trajectories. Problem formulation is shown to be a key step in this process. To illustrate the idea, a launch vehicle trajectory optimization problem is solved for maximizing the final velocity of the first stage of a multi-stage rocket assuming that all fuel will be expended. The sensitivity of the solution to uncertainties is examined by modeling environmental uncertainties as Gaussian processes in a Monte Carlo simulation. Combining optimal control and Monte Carlo analysis improves the planning process by allowing for worst case scenarios to be identified and mitigated.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

BVP	boundary value problem
DIDO	MATLAB optimal control toolbox
ECI	Earth centered inertial
GPS	global positioning system
HMC	Hamiltonian minimization condition
ISS	International Space Station
KSC	Kennedy space center
NWU	north-west-up
OCP	optimal control problem
ODE	ordinary differential equation
POST	program to optimize simulated trajectories
RLV	reusable launch vehicle

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

First, and foremost, I would like to thank my parents, Carl and Brigitte Manemeit, for their unwavering love and support over the years. I would not be where I am today if it wasn't for them. They have the ability to keep me going, even when times are tough. Second, I would like to thank my advisors, Dr. Mike Ross and Dr. Mark Karpenko, for their assistance in this process. Their guidance was instrumental in my accomplishment of this thesis. Next, I would like to thank all my friends that I have kept throughout the years. There are too many of you to name but you have all been the rock that I am able to fall back on. Lastly, I would like to thank the U.S. Navy for giving me the opportunity to study at the Naval Postgraduate School. I will cherish the education that I received here for the rest of my life.

THIS PAGE INTENTIONALLY LEFT BLANK

I.   INTRODUCTION

A.   OVERVIEW

Since the start of launching vehicles into space, there has been an ongoing effort to reduce the cost, safety, and reliability of a reusable launch vehicle (RLV). One aspect that this research is looking to correct is the time required to develop optimal launch trajectories. Optimal launch trajectories are essential to ensure that the most cost effective launch trajectory is flown. In this research, the algorithm that will be used is DIDO. DIDO is a MATLAB optimal control toolbox that was named after Dido, the founder and first queen of Carthage. She is famous for her use of mathematics in solving an optimal control problem (OCP) before calculus was even invented. DIDO is based on pseudospectral optimal control theory that is designed to solve an OCP in the same manner as using equations on a piece of paper [1]. The difficulties in solving for costates are eliminated by the convector mapping principle therefore DIDO produces spectrally accurate solutions [2]. With this tool, a more convenient method to determine launch trajectories can be developed to help reduce the time spent on the solution of a launch trajectory.

There is a great demand for satellite based equipment and it only keeps getting larger. The military is heavily reliant on launch vehicles since a vast majority of its net-centric warfare is based on satellite communications [3]. A great deal of U.S. national security surveillance is done via satellite for the ability to gather the most real time informational available [4]. The global positioning system (GPS) is not only vital to the military but to the civilian realm as well. The civil maritime and aviation communities rely heavily on GPS for accurate positioning for reliability and cost savings. Lastly, NASA has a huge demand for launch vehicles as they are responsible for resupplying the International Space Station (ISS) and sending probes for deep space exploration, as well as other satellite missions such as the James Webb Space Telescope [5].

B.   DIFFICULTIES WITH THE CURRENT APPROACH

The current cost to launch a pound of payload into an Earth orbit is around $10,000 [6]. In order to reduce the total cost of launch, industry is constantly looking to reduce the mass of the objects being sent into orbit. The other aspect is to reduce the cost to launch that object into space. This is the primary reason why companies are trying to optimize launch vehicle trajectories. The current industry standard for optimizing launch trajectories is the NASA program to optimize simulated trajectories (POST) [7]. This is an immensely complicated program that takes months to understand how to operate. POST takes the position of using a direct shooting method to calculate state variables as a function of time [8]. Another aspect is that POST requires an initial guess for each independent variable that would otherwise be held constant. Developing the initial guess can be very time consuming [9]. That complication leads to how long it takes to develop a launch trajectory and the intense man power required. A successful launch would require being able to predict conditions months in advance. If launch conditions are outside of those that were predicted, the launch may have to be terminated.

C.   OBJECTIVE

This thesis research was done to target the method in which launch trajectories are developed. The goal is to use a modern algorithm, DIDO, to reduce the time that is required to develop launch trajectories. DIDO removes the traditional shooting method to solve the OCP by using pseudospectral optimal control theory [1]. By being able to develop a trajectory closer to the launch date allows for a more accurate prediction of conditions to develop a more accurate trajectory. This will drastically reduce the manpower costs to become trained on the software and develop trajectories.

Another aspect that this thesis research contributes to is a move towards more automation. The goal being that the algorithm is robust enough that the only portions that need to be changed are the starting conditions and the endpoint conditions. This further increases the simplicity of the method to solve the given problem.

This thesis research specifically addresses the goal of maximizing the first stage final velocity. This problem was chosen in an effort to obtain a launch vehicle final

velocity that was closer to the final orbital velocity in a more expeditious manner. To account for real world uncertainties, a Gaussian process was used in a Monte Carlo simulation to allow the worst case performance to be identified. This knowledge will lead to more flexibility in the launch window and a more reliable launch trajectory.

D.   THESIS OUTLINE

This thesis is written in a manner that shows the reader the development of an optimal control problem to the application of optimal control to this research. Chapter II provides an introduction of optimal control and the process that is used to solve an optimal control problem. Chapter III introduces the launch problem that is to be solved by this thesis and also provides the hand calculations that set up the boundary value problem. These are used later for verification and validation of the pseudospectral optimal control solution. Chapter IV first starts off with a validation of the results to demonstrate that an optimal solution has been found. The chapter then displays the results for visualization of the trajectory and to prove that the results obtained were optimal using the derived equations from Chapter III and propagation of the controls. Chapter IV then introduces the uncertainty analysis that was performed to test for variations in environmental parameters. Chapter VI gives some conclusions and suggests some ideas for future work.

THIS PAGE INTENTIONALLY LEFT BLANK

II.   OPTIMAL CONTROL THEORY

A.   INTRODUCTION

The popularity for using optimal control is based on three main reasons. The first reason is that there is a cost function associated with the problem that can be minimized. The types of minimized cost can be time, fuel, effort, or any other performance objective. As stated before, the objective of this thesis research is to maximize final velocity. The next benefit to optimal control is the use of dynamics equations. The dynamics equations allow the user to more accurately model a trajectory that the system can fly. As compared to kinematics only, this allows for the prediction of what the system will do to a high degree of accuracy. Lastly, optimal control provides the ability to apply constraints to the system to be able to control the behavior. The constraints can be in the form of time, states, controls, and boundary conditions.

The process that is used to solve optimal control problems involves first constructing the Hamiltonian. The Hamiltonian is key to deriving the Hamiltonian minimization, the costate dynamics (adjoint) equations, and the transversality condition. In the following analysis, x is the state variable and u is the control variable. The costate vector used in this analysis is the Lagrange multiplier function and is annotated as $\lambda(t)$ [10]. The addition of this function will be further described in the next section.

B.   A GENERIC OPTIMAL CONTROL PROBLEM

A generic optimal control problem (OCP) is given as:

$$J\left[x(\bullet),u(\bullet)\right] = E\left(x\left(t_f\right),t_f\right) + \int_{t_0}^{t_f} F\left(x(t),u(t)\right)dt$$
$$\dot{x}(t) = f\left(x(t),u(t)\right)$$
$$x(t_0) = x^0$$
$$t_0 = t^0 \tag{1}$$
$$t_f = t^f$$
$$e\left(x(t_f)\right) = 0$$

In (1), $J\left[x(\bullet),u(\bullet)\right]$ is the cost function that should be minimized. The cost function is composed of the endpoint cost, $E\left(x(t_f),t_f\right)$ and the running cost, $\int_{t_0}^{t_f} F\left(x(t),u(t)\right)dt$. The endpoint cost is associated with the final time of the simulation. Example endpoint costs can be final time, remaining fuel, terminal velocity, etc. The running cost is cost accumulated during the entire flight time. An example of running cost is control effort. The dynamics portion is defined by $\dot{x} = f\left(x(t),u(t)\right)$, the initial condition is defined as x^0, and the start and end times are defined by t^0 and t^f. Lastly, any endpoint constraints are contained in the equation $e\left(x(t_f)\right) = 0$. An example of an endpoint constraint can be the conditions to maintain a specific orbit.
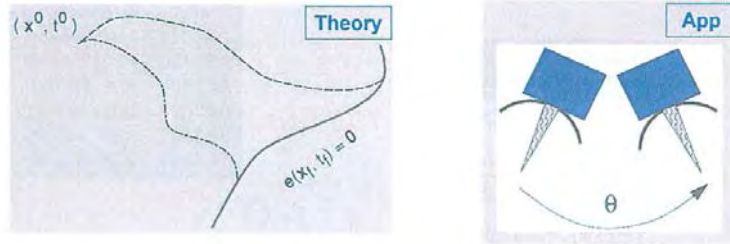

Figure 1.   Optimal maneuver, from [11]

The application of the above equations is seen in Figure 1 as the object starts at some initial condition and maneuvers along a trajectory to a desired end condition along a

given dynamic constraint. The problem starts at x^0, t^0 and the spacecraft maneuvers itself to endpoint, which satisfies the constraint of $e(x_f, t_f) = 0$

C.   SOLVING AN OPTIMAL CONTROL PROBLEM

In the 1950s, finding solutions to the standard problem formulation given in (1) was causing problems for Soviet engineers who knew that the problems being encountered were from the math and not the engineering. This led the Russian military to approach an individual by the name of Lev Pontryagin to help solve this problem. While creating a general problem of optimal control, Pontryagin realized that constraints on the control need to be included and special attention needs to be given to optimization. This gave birth to the present form of optimal control theory [12].

It is the minimization of the Hamiltonian that needs to be given the proper attention. When the Hamiltonian is minimized, the endpoints of the control constraint need to be evaluated to determine true minimum. This process is described as:

$$\min H(\lambda, x, u) \\ u^L \leq u \leq u^U \tag{2}$$

Equation (2) is the Hamiltonian minimization condition (HMC). Pontryagin proved that the minimized Hamiltonian is always constant as a function of time with the value zero for problems independent of time, -1 for a minimum time problem [12].

Solving Pontryagin's problem can be decomposed into four steps. When solving the problem, as stated above, the first step is to solve for the Hamiltonian. The Hamiltonian is a function of the running cost and the product of the costate vector with the dynamics equations:

$$H(\lambda, x, u) := F(x, u) + \lambda^T f(x, u) \tag{3}$$

The next step is to perform the Hamiltonian minimization. This involves taking the partial derivative of the Hamiltonian with respect to each control variable and setting the derivative equal to zero.

The goal of this step is to be able to remove the dependence of the control variable, u, from the Hamiltonian equation. If u does not appear explicitly, the partial

derivative is interpreted as a switching function. The switching function is describes how u switches from u_l to u_u (the control bounds) throughout the maneuver. The next step is to derive the adjoint equations. This step forms the dynamics of the costate variables, as a function of time, for each state variable. The adjoint equation is needed because the minimized Hamiltonian is a function of $\lambda(t)$. Therefore, the costate needs to be solved. This is done by taking the negative of the partial derivative of the Hamiltonian with respect to each of the state variable. This gives the adjoint equation.

$$\dot{\lambda}(t) = -\frac{\partial H}{\partial x} \tag{4}$$

The last step is to apply the transversality condition. This involves taking the partial of the endpoint Lagrangian with respect to the state at the terminal time.

$$\lambda(t_f) = \frac{\partial \overline{E}}{\partial x_f} \tag{5}$$

In (5) $\overline{E}$ is the Endpoint Lagrangian.

$$\overline{E}(\upsilon, x(t_f)) := E(x(t_f)) + \upsilon^T e(x(t_f)) \tag{6}$$

From here, the classical approach is to form a boundary value problem (BVP) using the dynamics and adjoint equations together with the boundary conditions and the transversality condition. A common approach to solve the BVP is to use a shooting method.

D.   EXAMPLE PROBLEM

To illustrate the application of the idea in the previous section, a simple example problem will be studied. The example that will be solved in the section is a 1-D linear quadratic problem [11].

$$J\left[x(\cdot), u(\cdot)\right] = \frac{1}{2}\int_{t_0}^{t_f} u^2(t)\, dt$$

$$\dot{x} = x + u$$

$$x_0 = 1$$

$$t_0 = 0 \tag{7}$$

$$t_f = 1$$

$$x_f = 0$$

Examining the cost function, it can be seen that $E\left(x(t_f)\right) = 0$ and $F(x,u) = \frac{1}{2}u^2$.

There is only one dynamics equation and that is $\dot{x} = f(x,u) = x + u$. From here, the Hamiltonian can now be derived.

$$H\left(\lambda, x, u\right) = F(x,u) + \lambda^T f(x,u) = \frac{1}{2}u^2 + \lambda(x+u) \tag{8}$$

Now, the Hamiltonian minimization is accomplished by taking the partial derivative of H with respect to u.

$$\frac{\partial H}{\partial u} = u + \lambda = 0 \tag{9}$$

This allows for u to be solved for in terms of λ, which gives $u = -\lambda$. Since u is a function of $\lambda(t)$, the adjoint equation will be needed to solve for the costate history. The adjoint equation is the next step in solving the problem. This involves taking the partial derivative of the Hamiltonian with respect to x.

$$-\dot{\lambda} = \frac{\partial H}{\partial x} = \lambda$$

$$\dot{\lambda} = -\lambda \tag{10}$$

From (10), it can be seen that the solution to the adjoint equation λ is an exponential. Now, the last part is to apply the transversality condition. The first part of this involves solving for the endpoint Lagrangian.

$$\overline{E}\left(\upsilon, x(t_f)\right) := E\left(x(t_f)\right) + \upsilon^T e\left(x(t_f)\right) = \upsilon x_f \tag{11}$$

Once the endpoint Lagrangian is constructed, to obtain the transversality condition, the partial derivative of the endpoint Lagrangian is taken with respect to the endpoint condition.

$$\lambda\left(t_f\right) = \frac{\partial \overline{E}}{\partial x_f} = \upsilon \tag{12}$$

The result in (12) shows that the value of the costate is an unknown, υ, at t_f. Thus no new information results from this step. The transversality condition is not necessary, in this case, because only two boundary conditions are needed and these are provided by the given problem. After these four steps are completed, the following boundary value problem can be constructed.

$$\begin{aligned}
\dot{x} &= x - \lambda \\
\dot{\lambda} &= -\lambda \\
x_0 &= 1 \\
x_f &= 0
\end{aligned} \tag{13}$$

From here, the problem can be solved numerically to obtain $\lambda(t)$ and hence $u(t)$, which is desired. This process is not necessarily easy to accomplish. Due to the instability of the Hamiltonian system, the integrated equation can "blow up" even in the face of a very accurate guess for the unknown initial values [13]. This is where the MATLAB tool DIDO can make life a lot easier. Once the cost function, dynamics equations, constraints, and events are programmed into DIDO, the algorithm will solve for the states, controls, Hamiltonian, and costates as a function of time, without the need to construct the BVP. This is much easier than building for the BVP and using a shooting algorithm to solve the problem. It takes away the need to build an algorithm that converges on a solution without an accurate initial guess. With the optimal control trajectories, they can be propagated to solve for the states via an ordinary differential equation (ODE) for verification and validation purposes.

E.   SUMMARY

This chapter explained why optimal control is widely used based on its many advantages. It then went on to set up a generic OCP that was to be solved using the method defined in this chapter. After the process for solving an OCP was defined, an example problem was introduced to further illustrate the procedure. The next chapter will define the launch vehicle problem that is to be addressed by this thesis research.

III. LAUNCH PROBLEM FORMULATION

A. INTRODUCTION

In this chapter, the launch vehicle problem is defined along with the desired goal. Here Pontryagin's principle is used to set up the BVP but it is not completely solved in this chapter. The BVP provides information that can be checked to verify that an optimal solution has been found. In the next chapter, the problem is solved using DIDO.

B. THE LAUNCH PROBLEM

This section will identify the variables and parameters that will be used to construct the launch OCP starting with states, controls, cost, and lastly the dynamics equations. The first part to building this problem is to define the state vector and the control vector. The state vector for the problem includes the Cartesian positions, velocities, and thrust direction cosines. They are shown in Equation 14.

$$X = \begin{bmatrix} x \\ y \\ z \\ v_x \\ v_y \\ v_z \\ c_x \\ c_y \\ c_z \end{bmatrix} \tag{14}$$

The positions (x, y, and z) are in units of kilometers (km), the velocities (v_x, v_y, and v_z) are in kilometers/second (km/s), and the thrust direction cosines (c_x, c_y, and c_z) are unit less as this unit vector that simply provides the direction of the constant thrust. The controls are the rates of change of the unit thrust vector and the Euclidean distance of the launch vehicle from the origin of a reference frame. The radius vector was added as a control to introduce a constraint to prevent the launch vehicle from entering the surface of the Earth, i.e. $r > r_E$. The constraint $r > r_E$ also avoids a potential situation where zero

would be in the denominator of the dynamics equation (see Equation 17). The control
vector is:

$$u = \begin{bmatrix} w_x \\ w_y \\ w_z \\ r \end{bmatrix}. \tag{15}$$

The goal for this problem is to maximize the final velocity of the launch vehicle and this
becomes the cost function. Final velocity was chosen to be the endpoint cost due to the
desire to achieve maximum velocity in the quickest manner possible. The bounds on time
that were used were a starting time of zero and a final time based on how long it took to
consume all first stage propellant. This allows for the launch vehicle to be closer to final
orbital velocity at first stage burnout. At a higher first stage burnout velocity, less thrust
input is required from the second stage to achieve the desired orbit. When performing the
analysis, the cost function is the variable that is to be minimized. In order to maximize
the final velocity, the cost function has to be the negative of the final velocity, which is
the same as minimizing the negative of final velocity seen in Equation 16. The square of
velocity was used to remove the need to have a square root in the equation. This
eliminates the potential of having a square root of zero, which has an infinite gradient.

$$J\big[x(\bullet), u(\bullet)\big] = -v_f^{\,2} \tag{16}$$

We now define the dynamics equations which will govern this problem. These equations
are formed from the time rate of change of the state variables. The dynamics are shown in
Equation 17.

$$
kinematics
\begin{cases}
\dot{x} = v_x \\
\dot{y} = v_y \\
\dot{z} = v_z
\end{cases}
$$

$$
dynamics
\begin{cases}
\dot{v}_x = \dfrac{T}{m}c_x - \dfrac{\mu}{r^3}x - \dfrac{\frac{1}{2}\rho v_{rel,x}^2 SC_d}{m} \\[2em]
\dot{v}_y = \dfrac{T}{m}c_y - \dfrac{\mu}{r^3}y - \dfrac{\frac{1}{2}\rho v_{rel,y}^2 SC_d}{m} \\[2em]
\dot{v}_z = \dfrac{T}{m}c_z - \dfrac{\mu}{r^3}z - \dfrac{\frac{1}{2}\rho v_{rel,z}^2 SC_d}{m} \\[2em]
\dot{m} = \dfrac{T}{I_{sp}g}
\end{cases}
\qquad (17)
$$

$$
unit\_thrust
\begin{cases}
\dot{c}_x = w_x \\
\dot{c}_y = w_y \\
\dot{c}_z = w_z
\end{cases}
$$

In (17), T is the thrust of the launch vehicle in N which is held constant during the flight time, m is the mass of the vehicle in kg, I_{sp} specific impulse of the vehicle in sec, μ is the gravitational constant of the earth in kg^3/s^2, ρ is the atmospheric density in kg/m^3, v_{rel}^2 is the relative velocity of the vehicle with the atmosphere in km/s, S is the surface area of the vehicle in m^2, and C_d is the coefficient of drag.

A path constraint was added to the problem in order to maintain the magnitude of the thrust direction cosines equal to one and the radius from the states x, y, and z is equal to the control radius. These constraints were necessary to ensure that the defined magnitude of thrust would not be exceeded and launch vehicle flight path remained outside the radius of the Earth. Than path constraints are given as:

$$
\begin{aligned}
c_x^2 + c_y^2 + c_z^2 - 1 &= 0 \\
x^2 + y^2 + z^2 - r^2 &= 0
\end{aligned}
\qquad (18)
$$

13

The full optimal control problem is now given as:

$$J[x(\cdot),u(\cdot)] = -v_f^{\,2}$$

$$\dot{x} = v_x$$

$$\dot{y} = v_y$$

$$\dot{z} = v_z$$

$$\dot{v}_x = \frac{T}{m}c_x - \frac{\mu}{r^3}x - \frac{\frac{1}{2}\rho v_{rel,x}^2 SC_d}{m}$$

$$\dot{v}_y = \frac{T}{m}c_y - \frac{\mu}{r^3}y - \frac{\frac{1}{2}\rho v_{rel,y}^2 SC_d}{m}$$

$$\dot{v}_z = \frac{T}{m}c_z - \frac{\mu}{r^3}z - \frac{\frac{1}{2}\rho v_{rel,z}^2 SC_d}{m}$$

$$\dot{m} = \frac{T}{I_{sp}g}$$

$$\dot{c}_x = w_x$$

$$\dot{c}_y = w_y$$

$$\dot{c}_z = w_z$$

$$x_0 = r_E \cos(Lat)\cos(Lon)$$

$$y_0 = r_E \cos(Lat)\sin(Lon)$$

$$z_0 = r_E \sin(Lat)$$

$$c_x^{\,2} + c_y^2 + c_z^2 - 1 = 0$$

$$x^2 + y^2 + z^2 - r^2 = 0$$

$$t_0 = 0$$

$$t_f = \frac{m_0 - m_f}{\dot{m}}$$

$$(19)$$

In (19), *Lat* and *Lon* are the latitude and longitude of the launch point and r_E is the radius of the Earth.

C. DEVELOPING THE BOUNDARY VALUE PROBLEM

As described in the previous chapter, the first step in setting up the OCP BVP involves solving for the Hamiltonian. Because the cost is only a function of final velocity, the running cost is zero and therefore $F(x,u)$ is zero. The Hamiltonian is now just a

14

function of the individual costates and time rate of change of each state variable.

$$H(\lambda,x,u) = \lambda_x v_x + \lambda_y v_y + \lambda_z v_z + \lambda_{vx}\left(\frac{T}{m}c_x - \frac{\mu x}{r^3} - \frac{1}{2}\rho v_{rel,x}^2 SC_d\right) +$$

$$\lambda_{vy}\left(\frac{T}{m}c_y - \frac{\mu y}{r^3} - \frac{1}{2}\rho v_{rel,y}^2 SC_d\right) + \lambda_{vz}\left(\frac{T}{m}c_z - \frac{\mu z}{r^3} - \frac{1}{2}\rho v_{rel,z}^2 SC_d\right) + \qquad (20)$$

$$\lambda_{u_x} w_x + \lambda_{u_y} w_y + \lambda_{u_z} w_z + \lambda_{path,u}\left(c_x^2 + c_y^2 + c_z^2 - 1\right) + \lambda_{path,r}\left(x^2 + y^2 + z^2 - r^2\right)$$

Now that the Hamiltonian is formed, the next part is to perform the Hamiltonian minimization. The partial derivative of H is performed with respect to each of the controls. The resulting relationships are given in (21).

$$\frac{\partial H}{\partial w_x} = \lambda_{u_x} = 0 := S_1$$

$$\frac{\partial H}{\partial w_y} = \lambda_{u_y} = 0 := S_2$$

$$\frac{\partial H}{\partial w_z} = \lambda_{u_z} = 0 := S_3 \qquad (21)$$

$$\frac{\partial H}{\partial r} = \frac{3\lambda_{v_x}\mu x}{r^4} + \frac{3\lambda_{v_y}\mu y}{r^4} + \frac{3\lambda_{v_z}\mu z}{r^4} - 2\lambda_{path,r}r = 0$$

As seen in (21), the Hamiltonian is linear in the thrust direction. Therefore, in accordance with Pontryagin's principle defined in the previous chapter, the partials need to be interpreted as switching functions shown by $S_1, S_2,$ and S_3. The adjoint equations are constructed next by taking the partial derivative of H with respect to the state vectors. The relationships are given by (22).

$$\dot{\lambda}_x = \frac{\partial H}{\partial x} = \frac{\lambda_{v_x}\mu}{r^3} - 2\lambda_{path,r}\,x$$

$$\dot{\lambda}_y = \frac{\partial H}{\partial y} = \frac{\lambda_{v_y}\mu}{r^3} - 2\lambda_{path,r}\,y$$

$$\dot{\lambda}_z = \frac{\partial H}{\partial z} = \frac{\lambda_{v_z}\mu}{r^3} - 2\lambda_{path,r}\,z$$

$$\dot{\lambda}_{v_x} = \frac{\partial H}{\partial v_x} = -\lambda_x$$

$$\dot{\lambda}_{v_y} = \frac{\partial H}{\partial v_y} = -\lambda_y$$

$$\dot{\lambda}_{v_z} = \frac{\partial H}{\partial v_z} = -\lambda_z$$

$$\dot{\lambda}_{u_x} = \frac{\partial H}{\partial c_x} = \frac{-\lambda_{v_x}T}{m} - 2\lambda_{path,u}c_x$$

$$\dot{\lambda}_{u_y} = \frac{\partial H}{\partial c_y} = \frac{-\lambda_{v_y}T}{m} - 2\lambda_{path,u}c_y$$

$$\dot{\lambda}_{u_z} = \frac{\partial H}{\partial c_z} = \frac{-\lambda_{v_z}T}{m} - 2\lambda_{path,u}c_z$$

(22)

For the transversality condition, the endpoint Lagrangian is based completely on the endpoint cost of maximizing final velocity.

$$\overline{E}\left(\upsilon, x(t_f)\right) = -\left(v_x(t_f)^2 + v_y(t_f)^2 + v_z(t_f)^2\right)$$

(23)

Now the partial derivative of the endpoint Lagrangian is performed for each of the velocities and those are shown in Equation 24.

$$\lambda_{v_x}(t_f) = \frac{\partial \overline{E}}{\partial v_x} = -2v_x(t_f)$$

$$\lambda_{v_y}(t_f) = \frac{\partial \overline{E}}{\partial v_y} = -2v_y(t_f)$$

$$\lambda_{v_z}(t_f) = \frac{\partial \overline{E}}{\partial v_z} = -2v_z(t_f)$$

(24)

In (24), it can be seen that the velocity costate endpoint is related to the final velocity. This can be useful as a verification and validation result. Similarly, in (22) the velocity

adjoint is a function of the position costate. This suggests that those costates may vary linearly which is also useful as a check during the verification and validation.

D. SUMMARY

This chapter defined the launch problem and showed how Pontryagin's principle can be used to construct the BVP for the launch problem. Once the BVP is constructed, it would be a very challenging process to obtain a solution using a shooting method (e.g., POST). The results obtained here will be used to verify a candidate solution. In the next chapter, DIDO is used to solve this problem with the goal of reducing computational time.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. LAUNCH TRAJECTORY OPTIMIZATION

A. INTRODUCTION

In this chapter, an optimal control solution is obtained with DIDO and verification and validation of the results is performed to indicate optimality of the solution. Then, the individual results will be displayed to analyze and illustrate the trends.

B. OPTIMIZATION RESULTS

The parameters that were used in this problem formulation can be seen in Table 1. After running DIDO, an optimal solution for 16 nodes was found indicating the problem was correctly posed. To confirm the results from the output of DIDO and series of plots were created for verification.

Parameter	Value/Range
m_0	219676 kg
m_f	6145 kg
I_{sp}	397.45 sec
T	960000 N
S	2.17 m^2
C_d	0.15
x, y, z	-6800 to 6800 km
v_x, v_y, v_z	-5 to 5 km/s, -5 to 5km/s, 0 to 5 km/s
c_x, c_y, c_z	-1 to 1, -1 to 1, 0 to 1
w_x, w_y, w_z	-0.1 to 0.1
r	6378.1363 to 6800 km
Lat	28°N 31 min 26.61 sec
Lon	80°W 39 min 3.06 sec

Table 1.   Model parameters, from [14]

After examining Figures 2 and 3, the output is what is desired in that there is a parabolic increase in the launch vehicle's velocity which obtains a final value of 4.73 km/s. The magnitude of the unit thrust vector is also constant at unity as required.
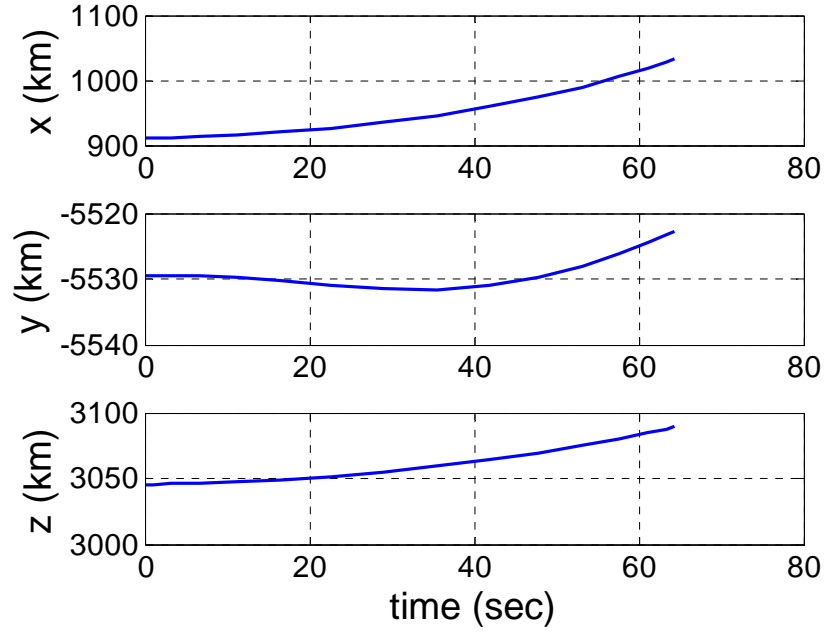

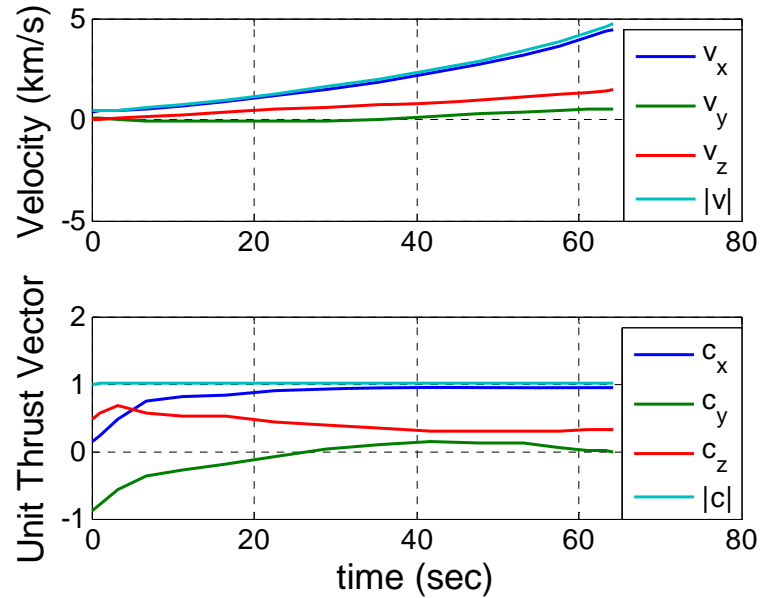Figure 2.   Position trajectories for maximum final velocity

Figure 3.   Velocity and unit-thrust vectors for maximum final velocity

After reviewing the plots of the costates from Figure 4, they behave as expected based on the adjoint equations that were derived from (22). Specifically examining the adjoint equations for the velocity costates, the derivative of the individual costate is the negative of the position costate.
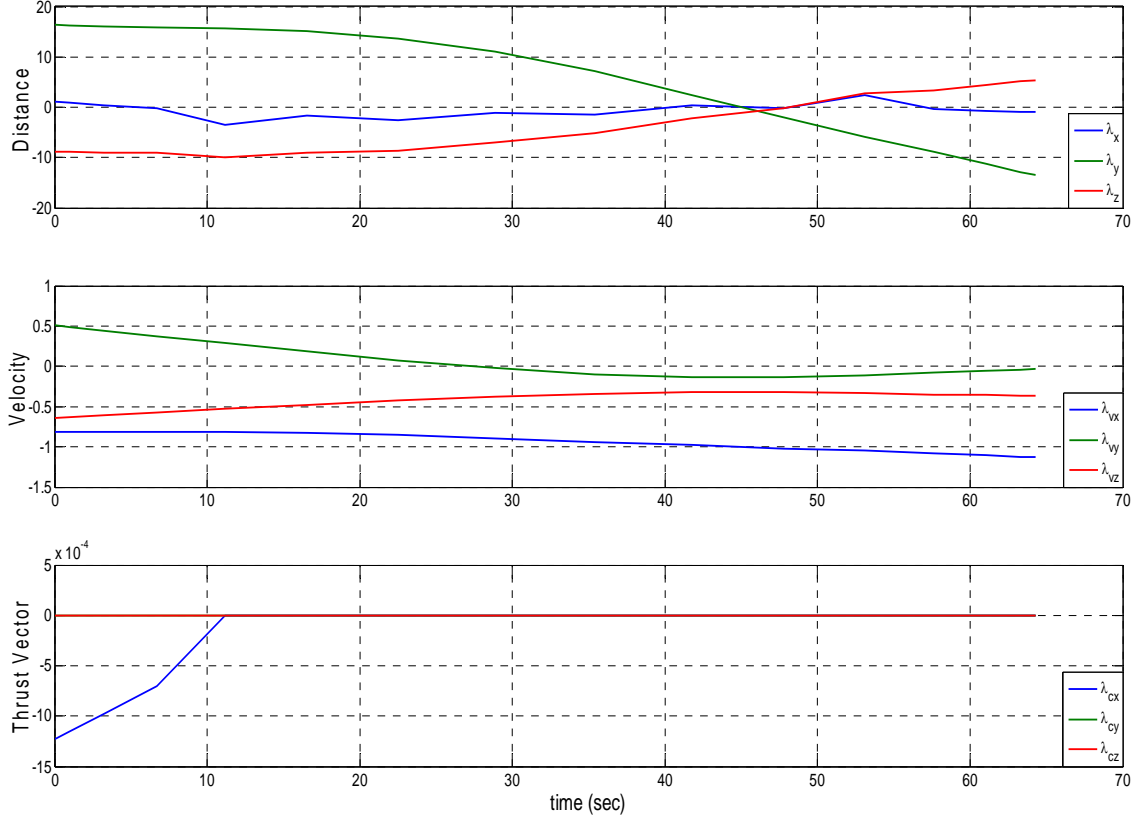
Figure 4.   Costate trajectories for maximum final velocity

In Figure 5, looking at λ_{v_z}, it starts with a positive slope and has a maximum around 45 seconds. The plot of $-\lambda_z$ starts off positive and crosses the y axis at the same time λ_{v_z} attains a maximum. The same can be done for the adjoint equations of v_x and v_y. As per (24), the final value of λ_{vx}, λ_{vy}, and λ_{vz} are required to be $-2v_x(t_f) = -8.92km/s$, $-2v_y(t_f) = -1.08km/s$, and $-2v_z(t_f) = -2.97km/s$, respectively, which is not the case after examining Figure 5. This inconsistency comes from the scaling that was used while

21

running DIDO. After removing the velocity scaling of 7.9054 km/s, the final velocity costate values agreed with the transversality conditions. This analysis further validates the results obtained by DIDO.
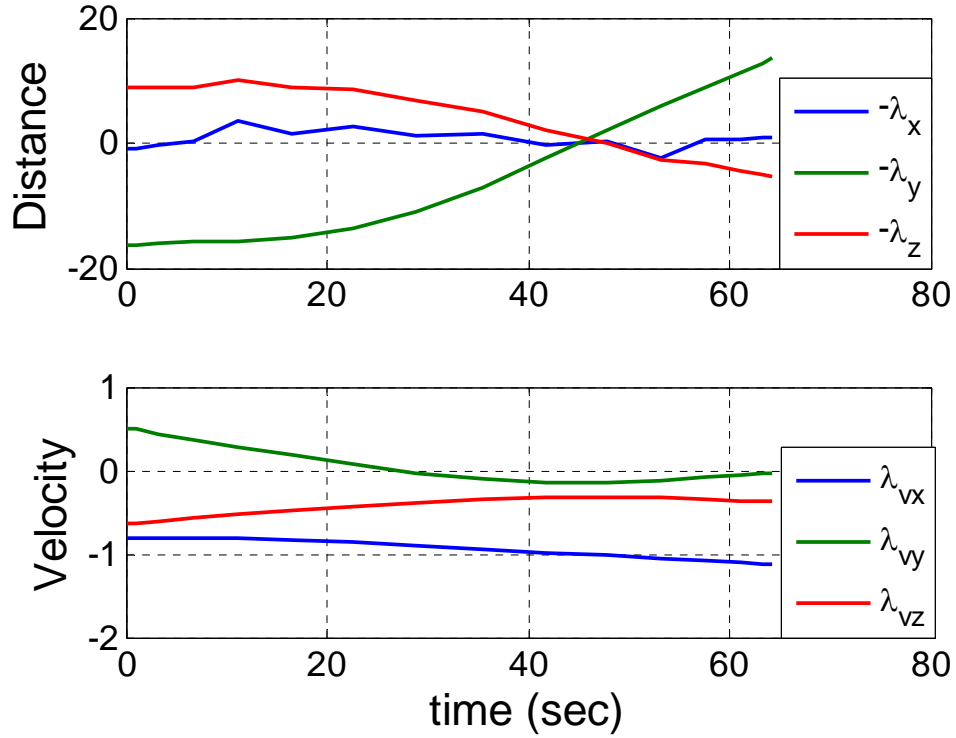

Figure 5.   Verification and validation of costates

Examining Figure 6, the altitude of the launch vehicle increases as the velocity increases and at burnout, attains an altitude of 33.59 km above the surface of the earth. The rapid rise in altitude is expected as the launch vehicle is at a constant thrust which is creating a linear rise in velocity coupled with an exponential decrease in atmospheric density reducing aerodynamic drag encountered by the launch vehicle.
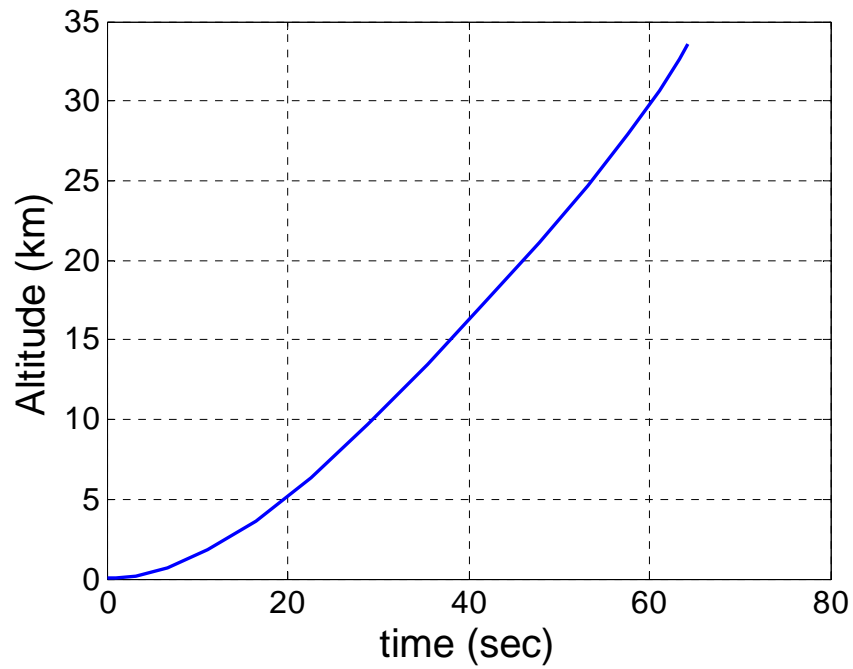
Figure 6.   Altitude as a function of time

Figure 7 represents the control vectors obtained from the solution. The radius was removed from the plot to be able to view the thrust rate of change more accurately. The thrust rate peaked at 0.1 at the very beginning which is the maximum defined by Table 1 but tapered off as time increased and eventually converged to zero. Since this is not a minimum time problem, the Hamiltonian constant is required to be zero. Looking at Figure 8 the Hamiltonian is approximately zero. This is indicative of an optimal solution.
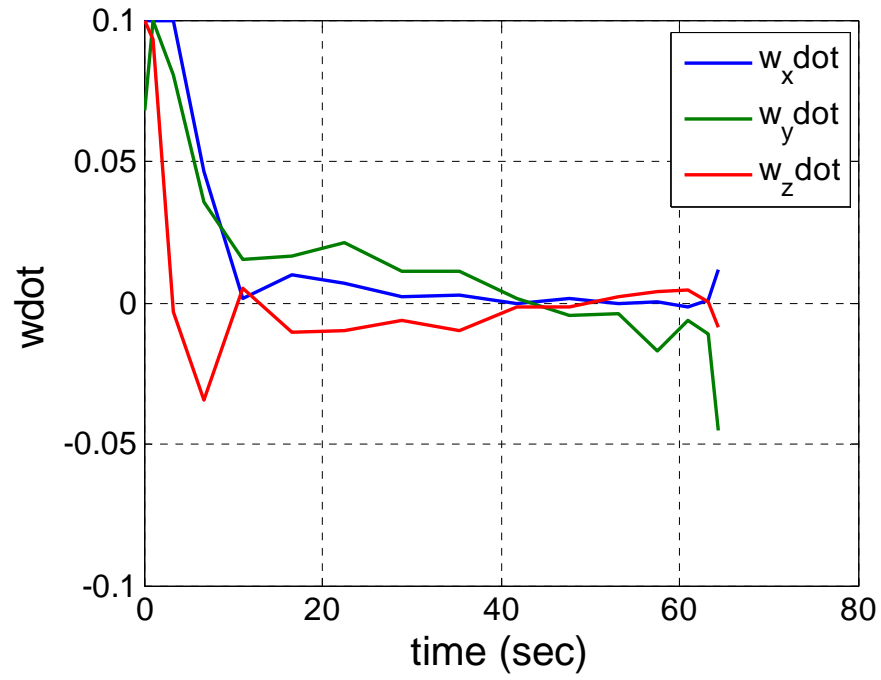
Figure 7.   Control vectors
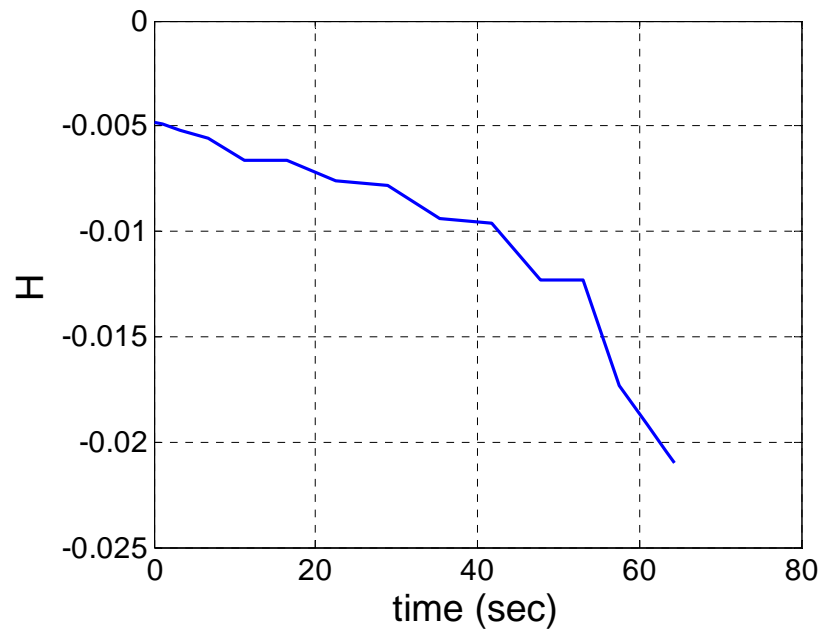
Figure 8.   Hamiltonian evolution as a function of time

C.   VERIFICATION AND VALIDATION

In order to verify the results that were obtained from DIDO, a simulation was run using the ode45 solver in MATLAB. The control vector which was obtained via DIDO is used to propagate the solution to verify the results. This will confirm whether or not the solution obtained via DIDO is feasible for implementation. When the two results are plotted against each other, it can be seen if the trends are the same or if there are large disparities in the data. If the two results are the same, it shows that the optimal solution obtained from DIDO is a valid one. If they are different, then the DIDO solution may not be accurate enough and the problem set up may need to be reevaluated and solved with larger number of nodes. Figure 9 shows the plots of the DIDO solution compared with the propagated solution
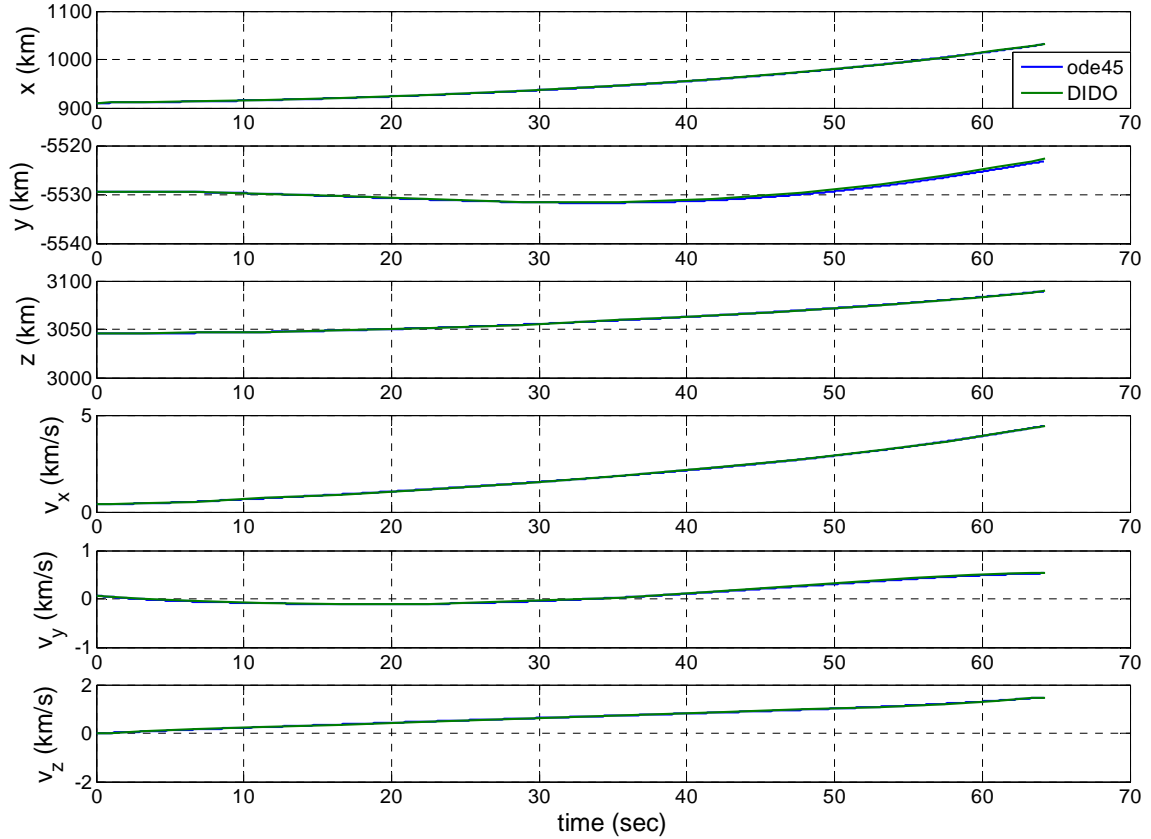

Figure 9.   Verification and validation of DIDO solution

In Figure 9, it can be seen that the two solutions are nearly identical. This shows that the solution obtained from DIDO is indeed a feasible one.

D. FURTHER ANALYSIS

After the optimal results were obtained from DIDO, there was some further analysis done to assist with better visualization of the trajectory. The first analysis was a coordinate transformation as seen in Figure 10. The simulation is best solved in ECI coordinates since most launches target a specific orbit and it is best to maintain the state vectors in a coordinate system that is centered intertially in the Earth. To better visualize the trajectory of the launch vehicle a rotation matrix was applied to the coordinates to transform them from Earth centered inertial to a north-west-up frame shown in Figure 10. This was done by rotating about the z-axis to position the y-axis at the longitude of the launch point. Next, the coordinate system was rotated about the new x-axis to point the z-axis to the latitude of the launch point. Lastly, the coordinate system was rotated about the new z-axis to point the x-axis in the north direction. The resulting coordinate system now has the x-axis pointing north, y-axis point west, and the z-axis as the zenith
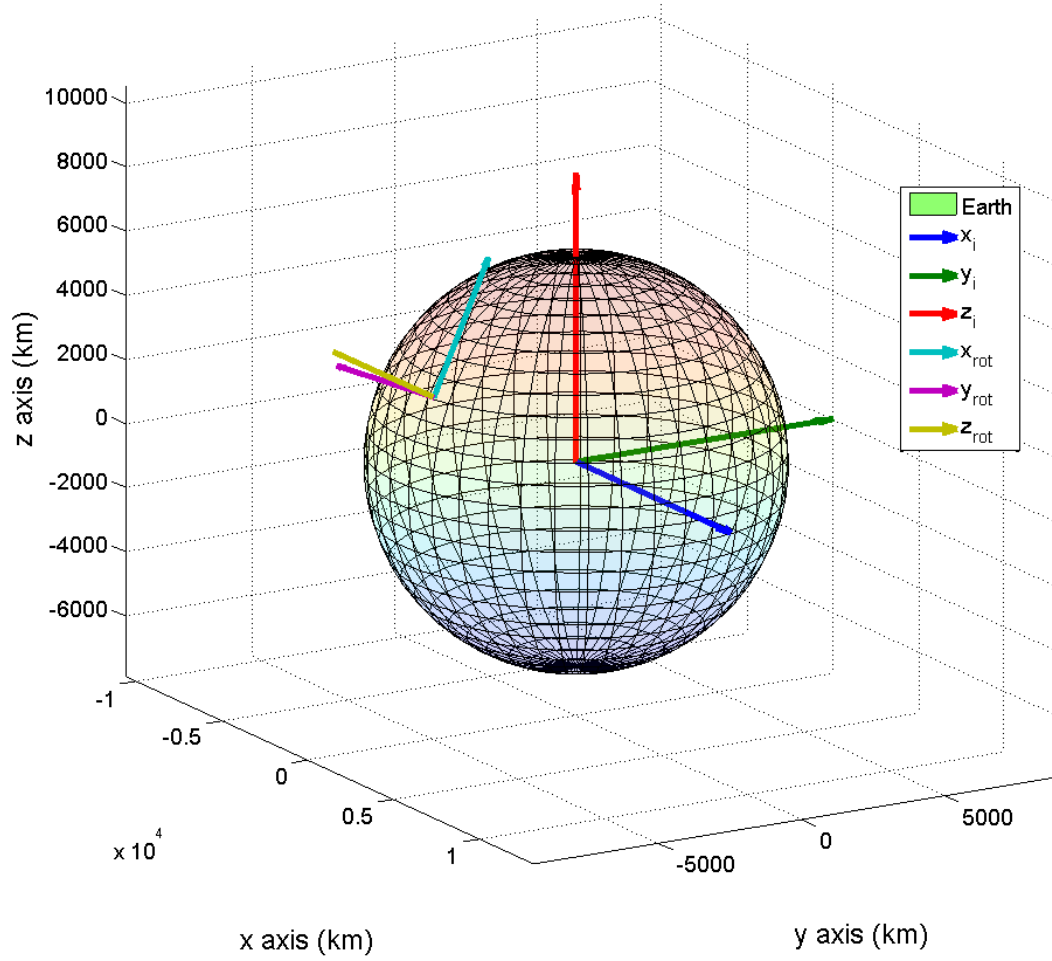
Figure 10.   Coordinate transformation from ECI to NWU

The transformation matrices used are given in (25) where *Lon* is the longitude of the launch site and *Lat* is the latitude.

$$C_1 = \begin{bmatrix} \cos\left(-\left(-Lon+\dfrac{\pi}{2}\right)\right) & \sin\left(-\left(-Lon+\dfrac{\pi}{2}\right)\right) & 0 \\ -\sin\left(-\left(-Lon+\dfrac{\pi}{2}\right)\right) & \cos\left(-\left(-Lon+\dfrac{\pi}{2}\right)\right) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$C_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\left(-\left(\dfrac{\pi}{2}-Lat\right)\right) & \sin\left(-\left(\dfrac{\pi}{2}-Lat\right)\right) \\ 0 & -\sin\left(-\left(\dfrac{\pi}{2}-Lat\right)\right) & \cos\left(-\left(\dfrac{\pi}{2}-Lat\right)\right) \end{bmatrix}$$

$$C_3 = \begin{bmatrix} \cos\left(-\dfrac{\pi}{2}\right) & \sin\left(-\dfrac{\pi}{2}\right) & 0 \\ -\sin\left(-\dfrac{\pi}{2}\right) & \cos\left(-\dfrac{\pi}{2}\right) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{25}$$

After substituting the *Lon* and *Lat* for KSC, we obtain a final rotation matrix.

$$C = C_3 C_2 C_1 = \begin{bmatrix} -0.0776 & 0.4712 & 0.8786 \\ -0.9867 & -0.1625 & 0 \\ 0.1427 & -0.8669 & 0.4775 \end{bmatrix} \tag{26}$$

Figures 11 and 12 show the launch vehicle's trajectory after performing the coordinate transformation with the origin of the new coordinate system being the launch point. The results are exactly as expected in that initially the thrust vector is vertical direction and beginning to turn over into the direction of flight. That is consistent with the constraint that was applied to ensure that the initial thrust vector was aligned with launch point's radius. The velocity is at all times tangential to the launch vehicle's trajectory.
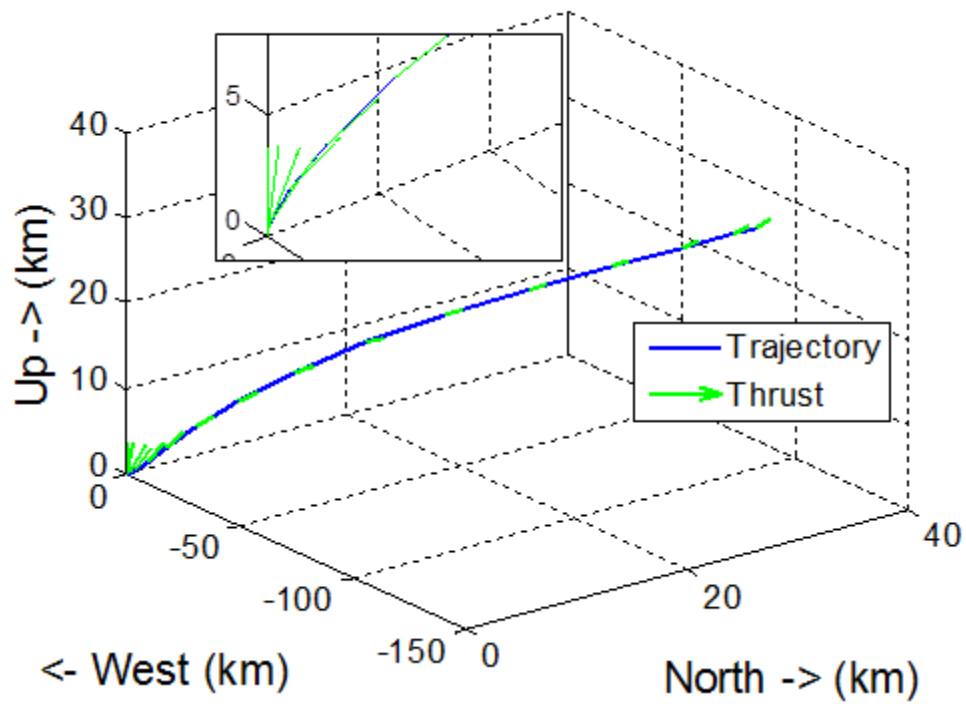
Figure 11.   Launch vehicle trajectory with thrust vectors
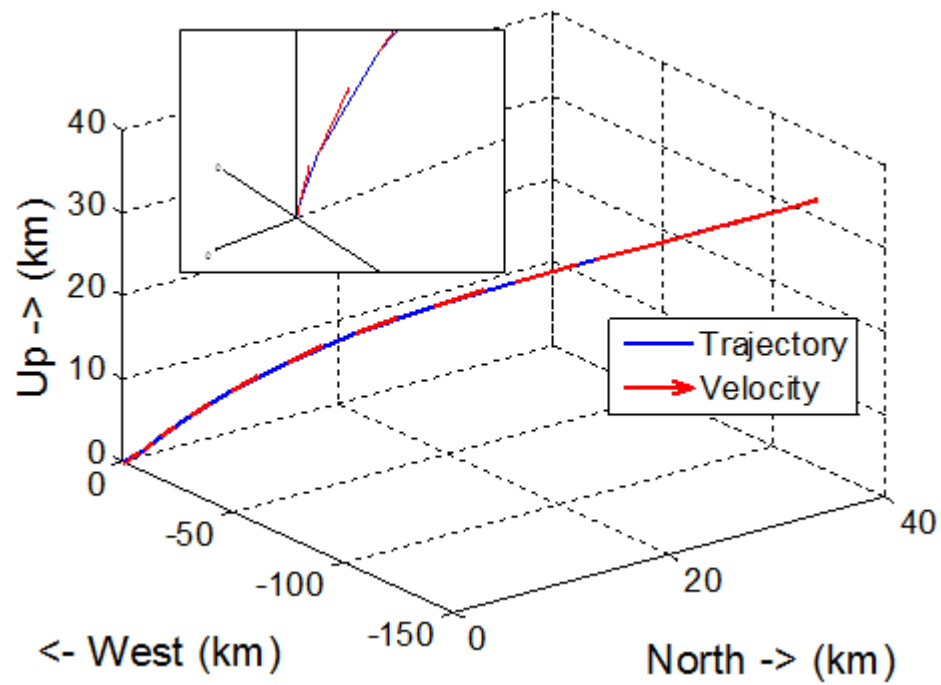

Figure 12.   Launch vehicle trajectory with velocity vectors

The next analysis performed was to plot the rotated trajectory on a Google Earth plot to show how the trajectory performed when plotted in reference to the land mass. Figures 13 and 14 provide a perspective of the launch vehicle as it leaves the launch point and travels in a north easterly direction. This is consistent with launches that are currently done at the KSC in that once the launch vehicle leaves the launch pad, it heads in a north easterly direction to ensure a safe area to drop the boosters. Figure 15 shows a plot of the STS-135 launch [15]. It can be clearly seen that the trajectory developed using DIDO is very similar to the trajectories used by NASA for the space shuttle (shown by the trajectory given in Figure 15).
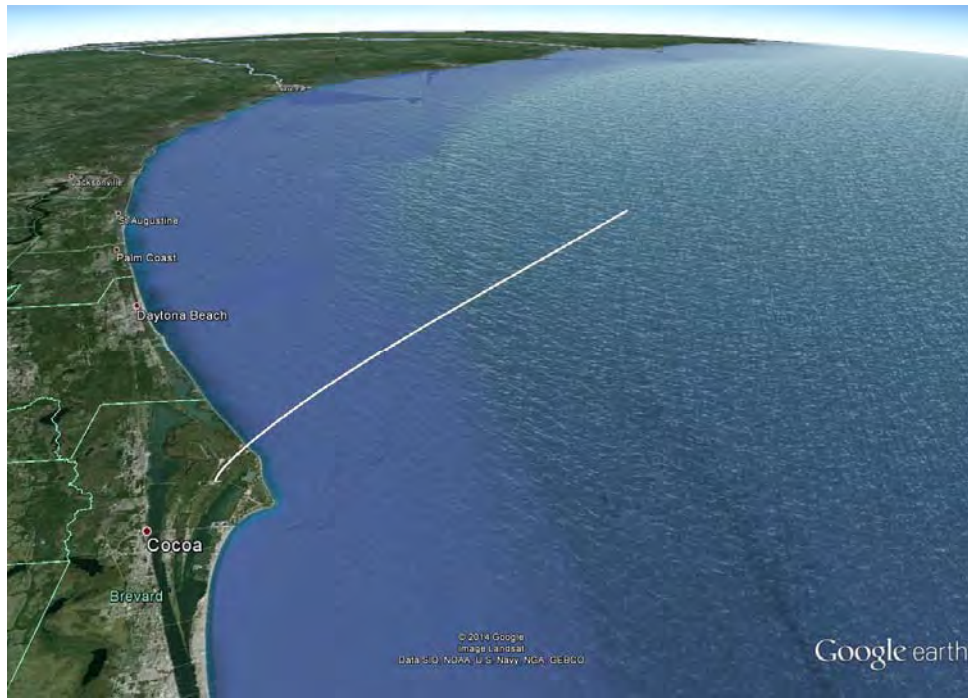

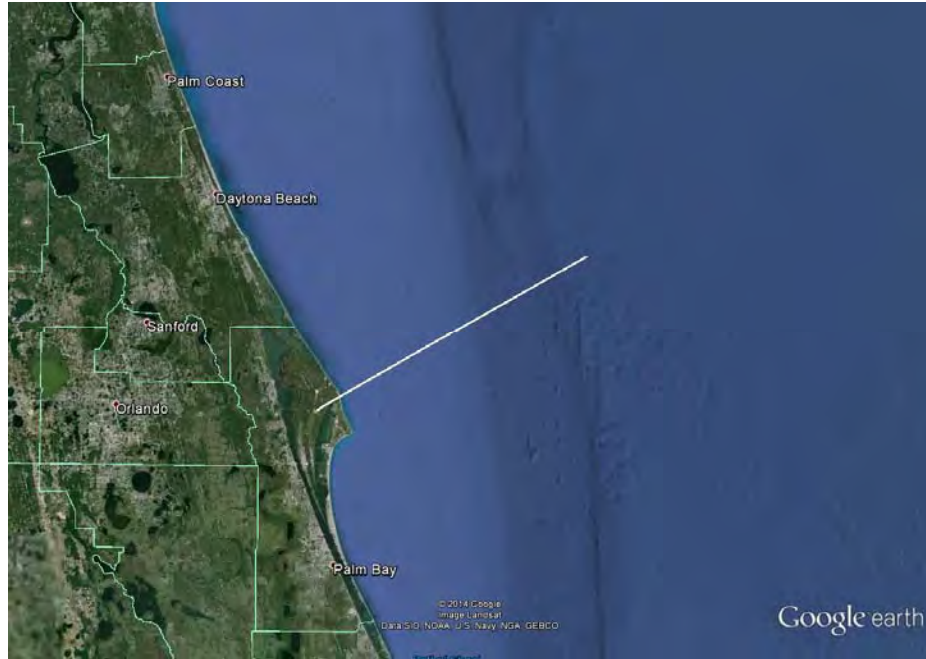Figure 13.   Google earth 3D view of the launch trajectory

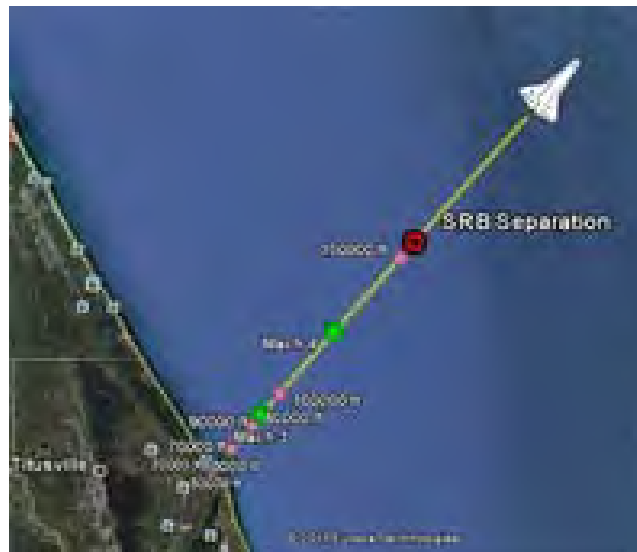Figure 14.   Google earth 2D view of the launch trajectory

Figure 15.   Google earth 2D view of STS-135 trajectory, from [15]

E.   PROBLEMS ENCOUNTERED

Throughout the process of creating a solution to this problem, there were many hurdles that had to be overcome in order to produce viable results. The first challenge was establishing a type of coordinate system that was to be used in order to model the

31

launch problem. First, the problem formulation was modeled in Cartesian coordinates. This created problems in effectively being able to maintain the trajectory from colliding with the surface of the Earth. The problem formulation was then shifted to polar coordinates which allowed for maintaining the radius of the trajectory outside the surface of the earth. The solution was then obtained in drag free environment but it was still desired to keep the problem in Cartesian coordinates. The problem was then shifted back to Cartesian and an optimal solution was finally obtained for a drag free environment. Once drag was introduced it was becoming impossible to keep the launch vehicle from colliding with the surface of the earth. After extensive isolation of the components to the dynamics equation, it was determined that the equation in which density was being calculated was the source of the problem. Once that was isolated, an optimal solution was found which lead to creating the method of solving for density that will be mentioned in the next chapter. Overcoming these challenges emphasizes that proper problem formulation is critical to successfully solving the problem.

F.   SUMMARY

In this chapter, the optimal control solution for launch was presented and evaluated for feasibility. The validation checks were accomplished using the derived equations from the previous chapter. Once those checks were complete, the controls obtained via DIDO were propagated to obtain a new set of state variables. The two sets of states were plotted against each other to establish feasibility of the solution. The two results were nearly identical proving the solution was feasible. Next, the trajectory was transformed to another coordinate frame for visual reference and plotted using a Google Earth to evaluate how the trajectory performs with land mass visible. Once again that was checked against trajectories flow by NASA for the space shuttle. The present solution is very similar to existing trajectories.

V.   MONTE CARLO SIMULATION

A.   INTRODUCTION

One way to assess the impact of uncertainties in a dynamic system is by the use of a Monte Carlo analysis. This is done by drawing from a large pool of random samples and observing their behavior [16]. This chapter starts off by describing the density model used to model atmospheric density as a function of temperature offset and altitude. Next the Monte Carlo simulations that were performed to assess the effects of uncertainties in launch environmental are described. Three Monte Carlo simulation studies were performed. The first was a temperature only simulation, the second was a wind only simulation, and the third was a combination of both temperature and wind. Temperature and wind were chosen based on the possibility of these effects having the largest influence on the launch vehicle.

B.   DENSITY MODELING

One model predicting atmospheric density as a function of altitude, r, uses an exponential form [14] .

$$\rho(r) = \rho_0 \exp\left(\frac{r_0 - r}{H}\right) \tag{27}$$

In (27), ρ_0 is the atmospheric density at sea level and H is the scale height parameter. For the density analysis in this thesis, various data points were taken from the 1976 Standard Atmosphere in order to calculate density as a function of altitude (*alt*) and temperature [17]. Data points were taken for temperature offsets (*TO*) from -30°C to 30°C in 10°C increments. Once those points were obtained they were plotted on an Excel scatter plot and a sixth order polynomial was used to create a curve fit of density versus altitude for each TO as seen in Figure 16. The format of the equation for modeling density is given as

$$\rho = a_1 alt^6 + a_2 alt^5 + a_3 alt^4 + a_4 alt^3 + a_5 alt^2 + a_6 alt + a_7 \tag{28}$$

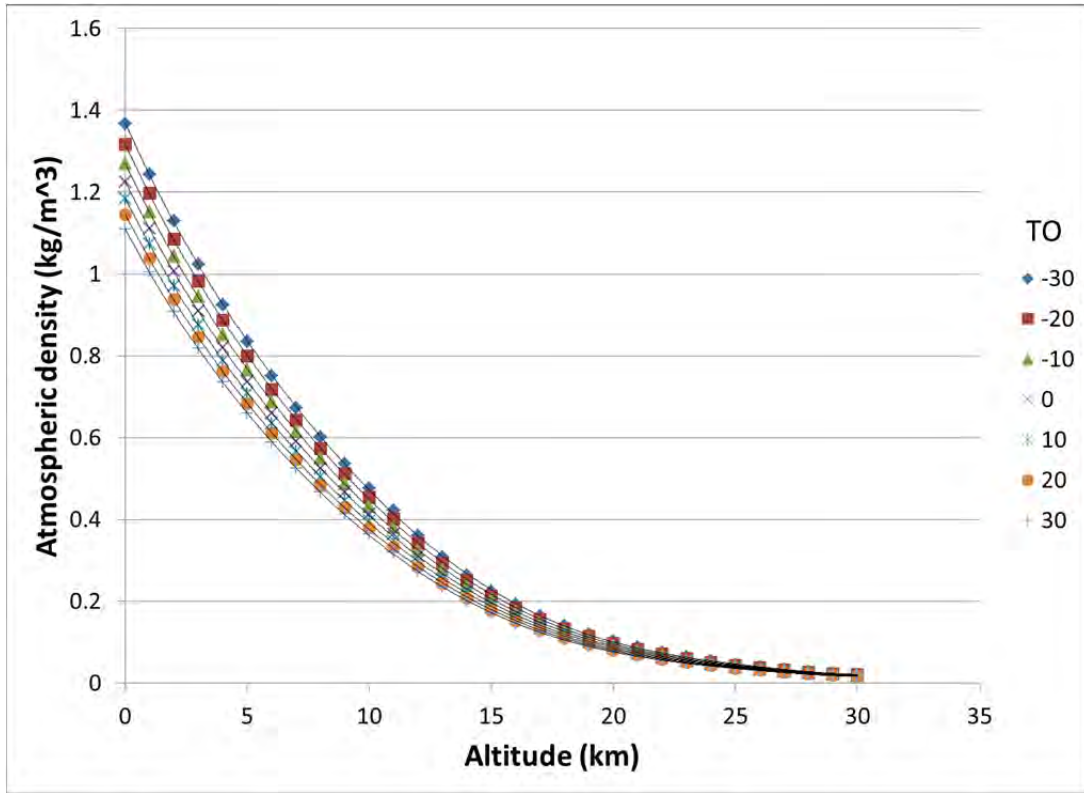Table 2 shows the coefficients at each power of *alt* for the given *TO*.

Figure 16.   Air density as a function of altitude and temperature

TO (°C)	a_1	a_2	a_3	a_4	a_5	a_6	a_7
-30	$9.482e^{-9}$	$-9.176e^{-7}$	$3.329e^{-5}$	$-5.958e^{-4}$	$8.025e^{-3}$	-0.1355	1.370
-20	$8.699e^{-9}$	$-8.380e^{-7}$	$3.028e^{-5}$	$-5.444e^{-4}$	$7.587e^{-3}$	-0.1312	1.316
-10	$8.007e^{-9}$	$-7.680e^{-7}$	$2.765e^{-5}$	$-4.996e^{-4}$	$7.199e^{-3}$	-0.1271	1.271
0	$7.398e^{-9}$	$-7.066e^{-7}$	$2.536e^{-5}$	$-4.607e^{-4}$	$6.855e^{-3}$	-0.1233	1.227
10	$6.855e^{-9}$	$-6.523e^{-7}$	$2.333e^{-5}$	$-4.266e^{-4}$	$6.546e^{-3}$	-0.1197	1.186
20	$6.373e^{-9}$	$-6.042e^{-7}$	$2.155e^{-5}$	$-3.965e^{-4}$	$6.269e^{-3}$	-0.1163	1.147
30	$5.944e^{-9}$	$-5.615e^{-7}$	$1.998e^{-5}$	$-3.700e^{-4}$	$6.019e^{-3}$	-0.1131	1.111

Table 2.   Coefficients from density curve fits

Each of the columns of coefficients given in Table 2 were also fit to a linear curve to be able to be able to compute each coefficient as a function of *TO*. This allows a single

34

equation for density to be developed as a function of *alt* and *TO*. Figures 17 through 23 show the curve fits for the given values of *TO*, for each of the coefficients in Table 2.
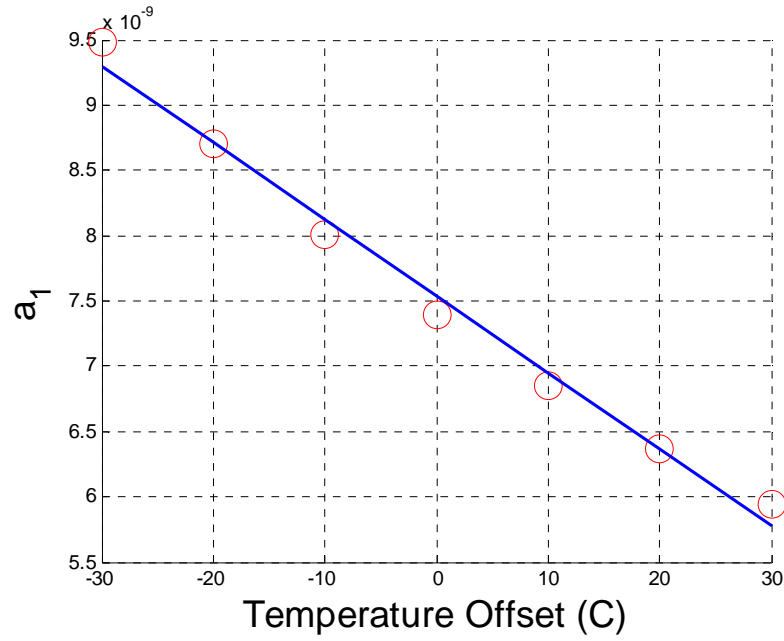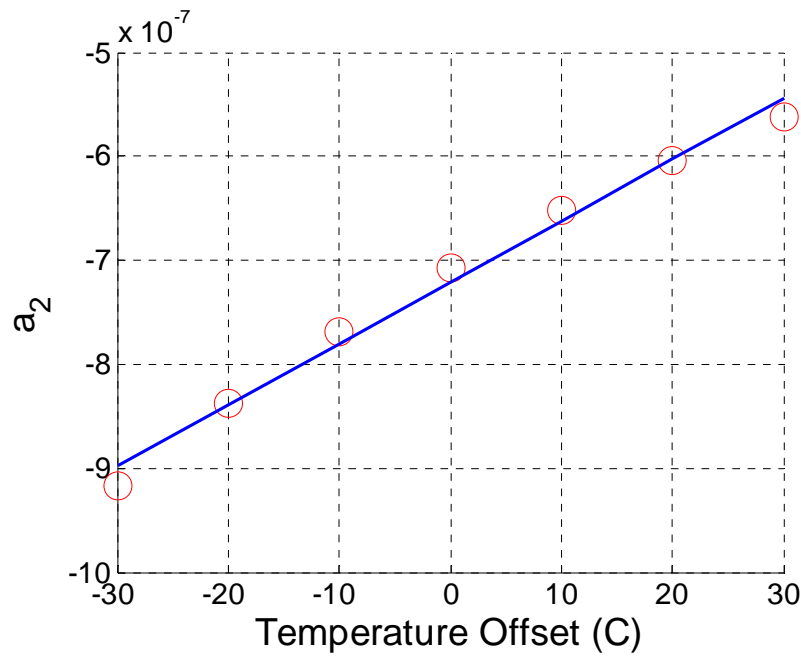

Figure 17.   Curve fit for a_1
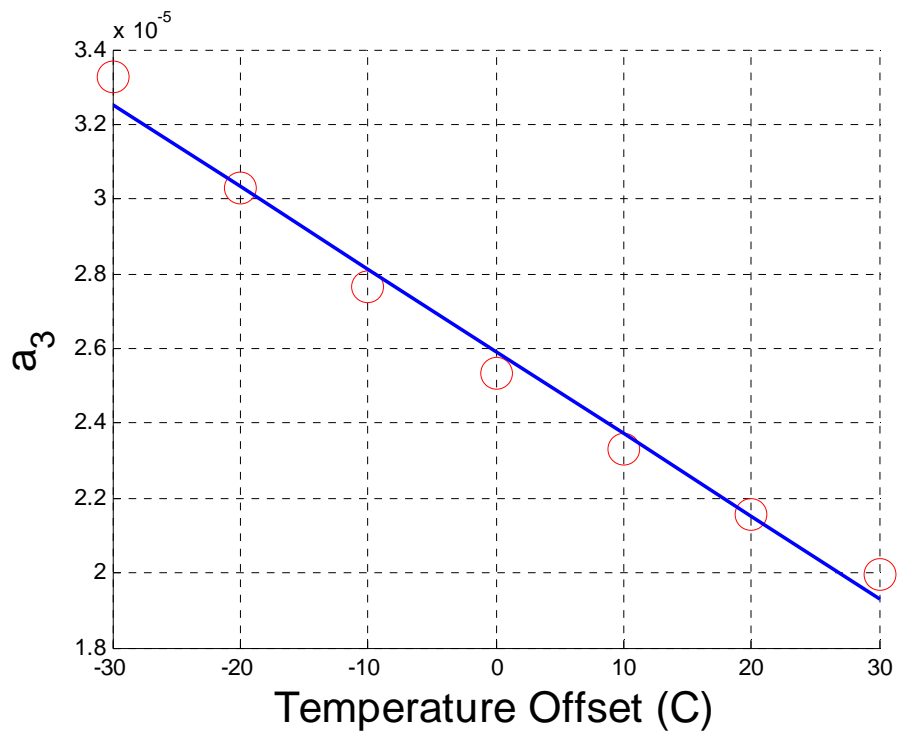

Figure 18.   Curve fit for a_2
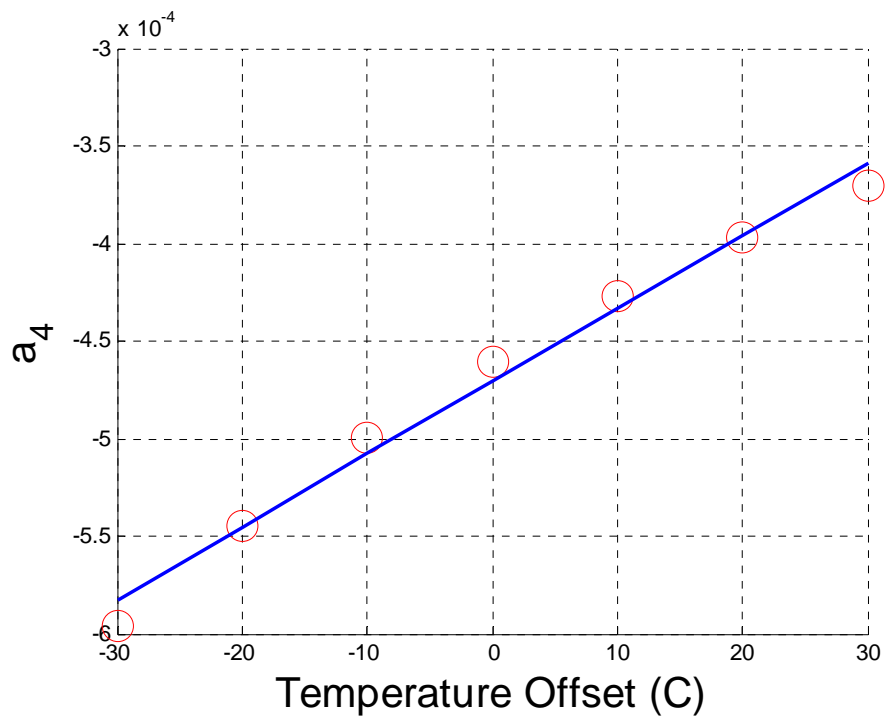
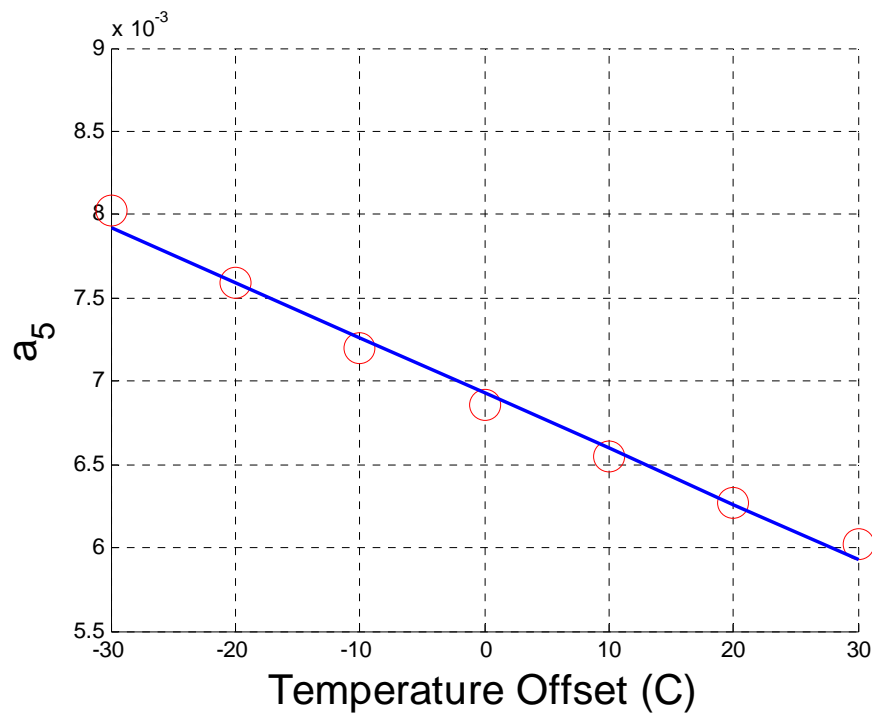Figure 19.   Curve fit for a_3

Figure 20.   Curve fit for a_4
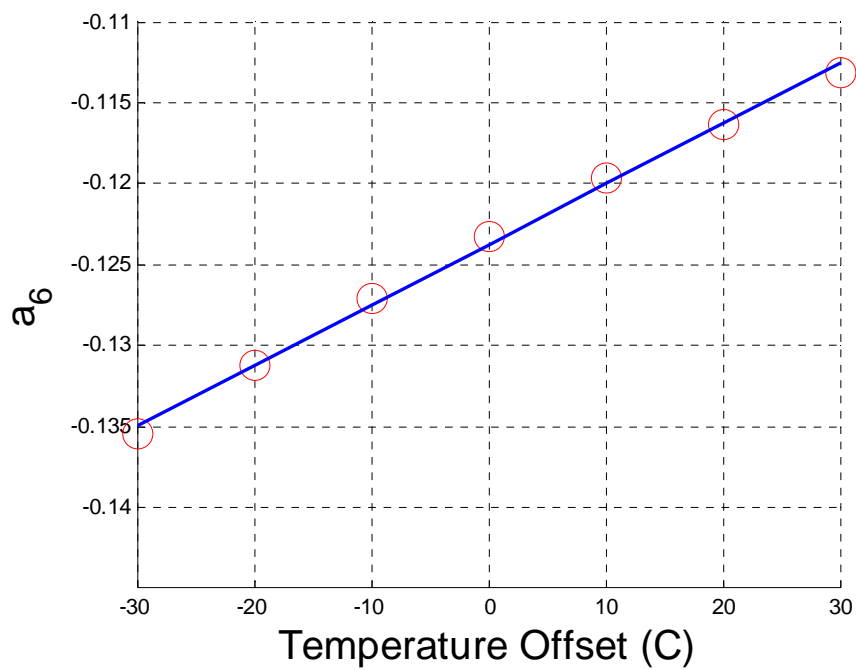
36

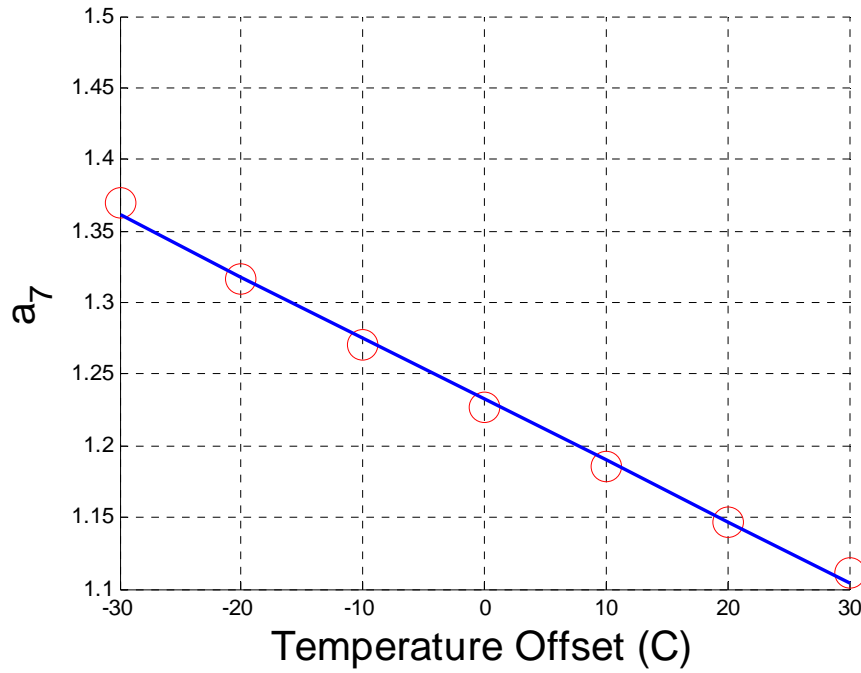Figure 21.   Curve fit for a_5

Figure 22.   Curve fit for a_6

Figure 23.   Curve fit for a_7

Using the results from Figures 17 through 23, a single equation can be developed for calculating density:

$$\rho(alt, TO) = \left(-5.864e^{-11} * TO + 7.537e^{-9}\right) * alt^6 + (5.899e^{-9} * TO - 7.212e^{-7}) * alt^5 +$$
$$(-2.204e^{-7} * TO + 2.592e^{-5}) * alt^4 + (3.736e^{-6} * TO - 4.705e^{-4}) * alt^3 + \tag{29}$$
$$(-3.324e^{-5} * TO + 6.929e^{-3}) * alt^2 + (3.729e^{-4} * TO - .1237) * alt - 4.307e^{-3} * TO + 1.233$$

Figure 24 shows the error of the single equation (29) and exponential density (27) with the densities obtained from the 1976 standard atmosphere.
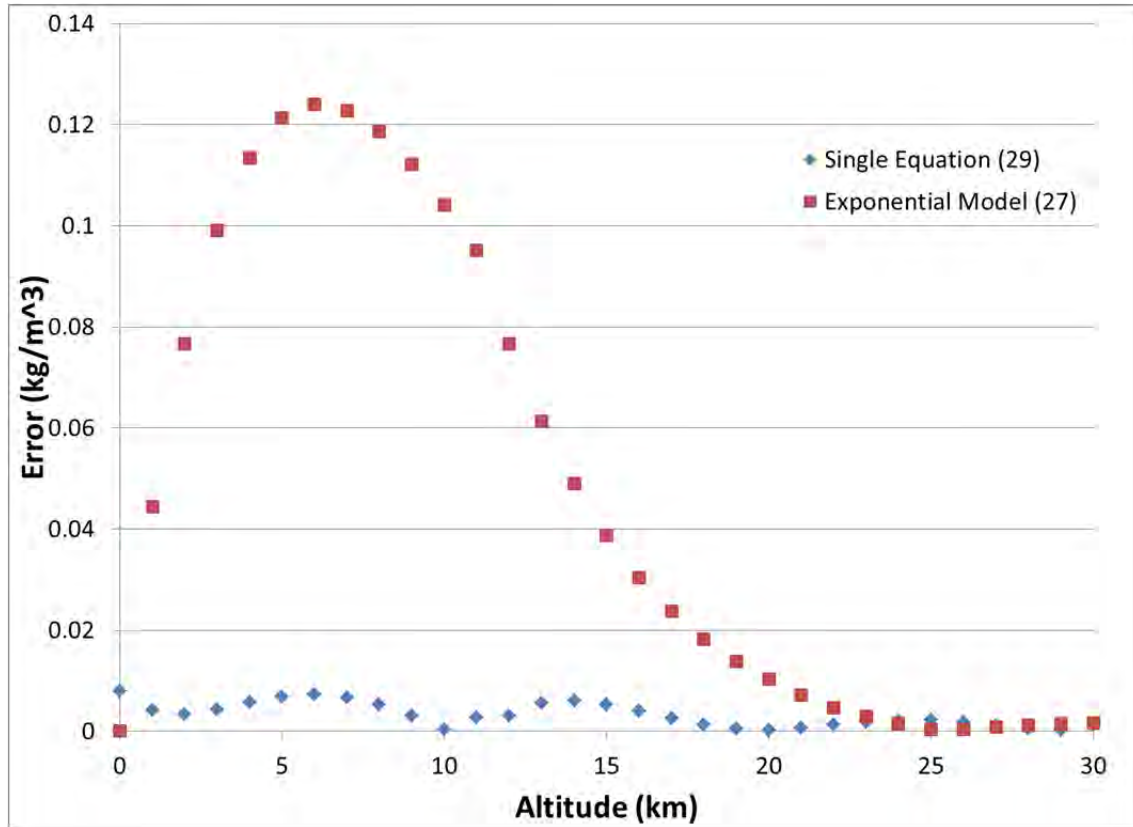
Figure 24.   Error comparison of single equation and exponential for *TO=0°C*

The error corresponding to the model (29) maintains a relatively constant value, close to zero whereas the exponential model has a rather high error during the first two thirds of the flight regime. Using the single equation that was produced by this thesis, the air density calculated will be more accurate and provide a better estimate of atmospheric drag during flight. Accordingly, this was the model used for obtaining the optimal launch trajectory discussed in the last chapter.

C.   TEMPERATURE VARIATION

After the solution from DIDO was validated (see Chapter IV), the next step is to add a certain degree of variation into the dynamics so the effects can be analyzed. The first simulation analyzed the effects of temperature variation. A Monte Carlo simulation was run for 1000 different points using a normal temperature distribution to introduce the uncertainty. The 1-σ variation used in the temperature simulation was 30°C. Varying the

temperature changes the air density the launch vehicle encounters, which changes the amount of drag felt on the launch vehicle. The equation that was used to model temperature variation is given by (30).

$$TO = TO_{norm} + \sigma_{temp} * n \qquad (30)$$

In (30), *TO* is the random temperature used in the density calculation, $TO_{norm} = 0°C$ is the temperature offset based on current conditions, $\sigma_{temp} = 30°C$ is the temperature variation, and *n* is a random number produced from a normal distribution with $\mu = 0$ and $\sigma = 1$. Figure 25 shows the resulting trajectories from the Monte Carlo simulation while Figure 26 shows the endpoints of the trajectories in the north-up plane. The variation in the north-west plane is shown in Figure 27. Referring to Figures 25 through 27, it appears that the launch trajectory is quite insensitive to large variation in temperature and the desired trajectory can still be achieved.
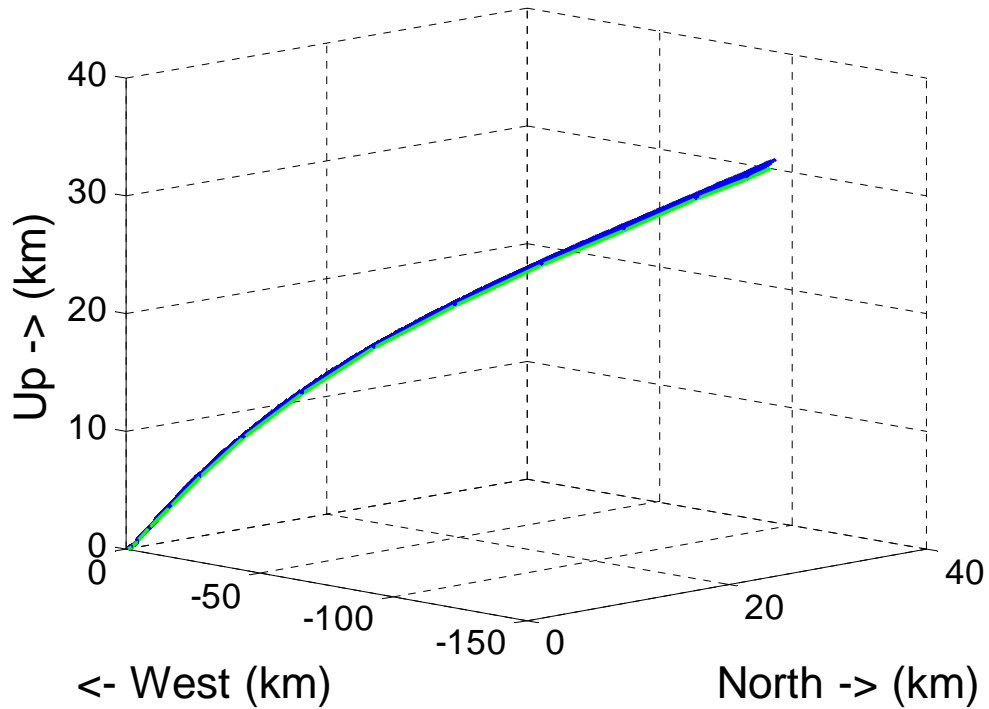

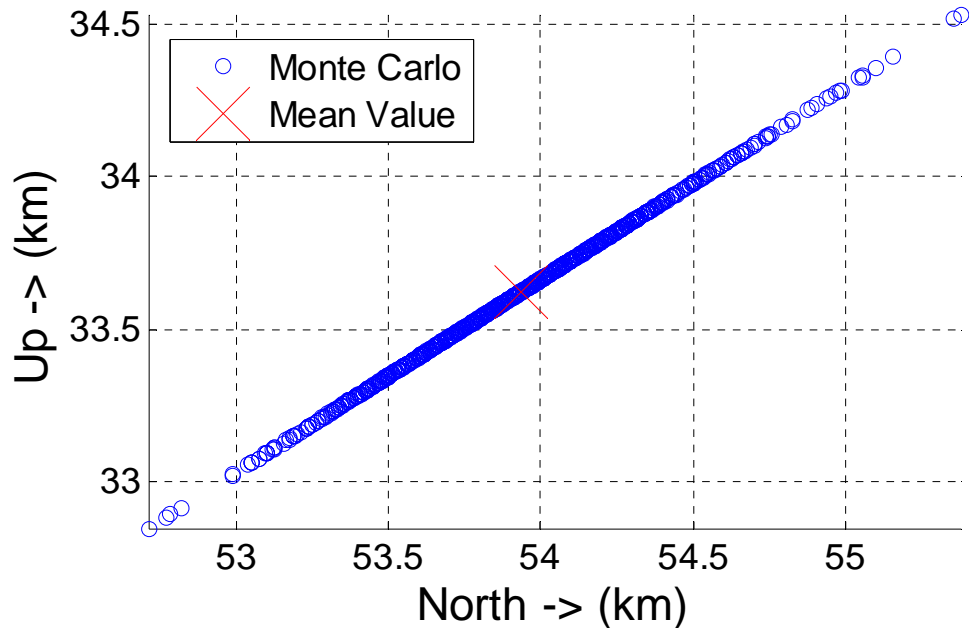Figure 25.   Monte Carlo simulation for temperature uncertainty

Figure 26.   Plot of endpoints from Monte Carlo for temperature in North-Up
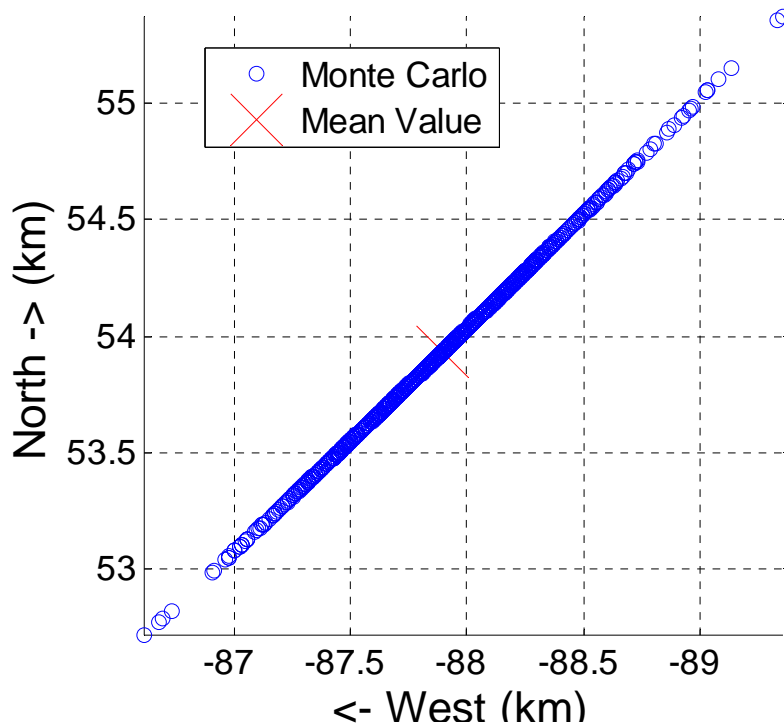

Figure 27.   Plot of endpoints from Monte Carlo for temperature in North-West

A plot of the distribution of temperature was created to verify that a correct distribution was being used during the Monte Carlo simulation. That distribution is seen

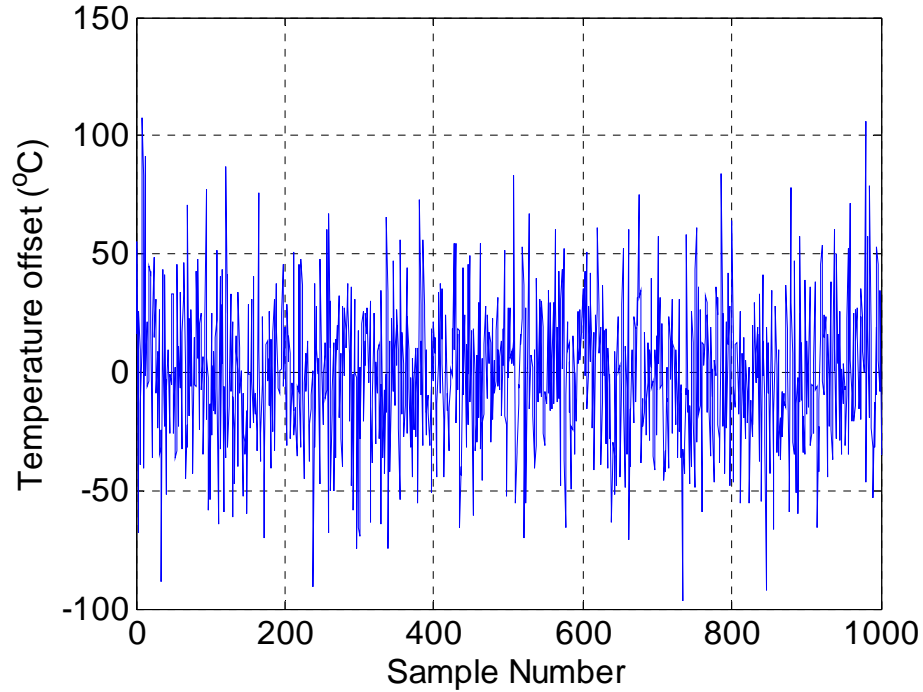in Figure 28. The distribution looks as expected with a mean value of zero and a standard deviation of 30°C.

Figure 28. Temperature offsets for Monte Carlo simulation

D. WIND VARIATION

The next part of the Monte Carlo simulation involved introducing a wind variation. The expected wind patterns were obtained from the NOAA Earth Systems Research Laboratory (ERSL) for the periods of January to December at a level of 300 mb which is equivalent to 30,000 ft. [18]. Figures 29 and 30 are the graphics obtained from ERSL website [18].
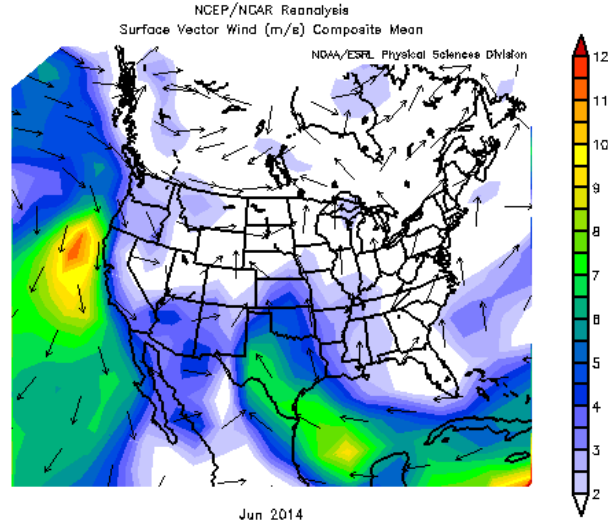
Figure 29.   Wind directions over North America at the surface, from [18]
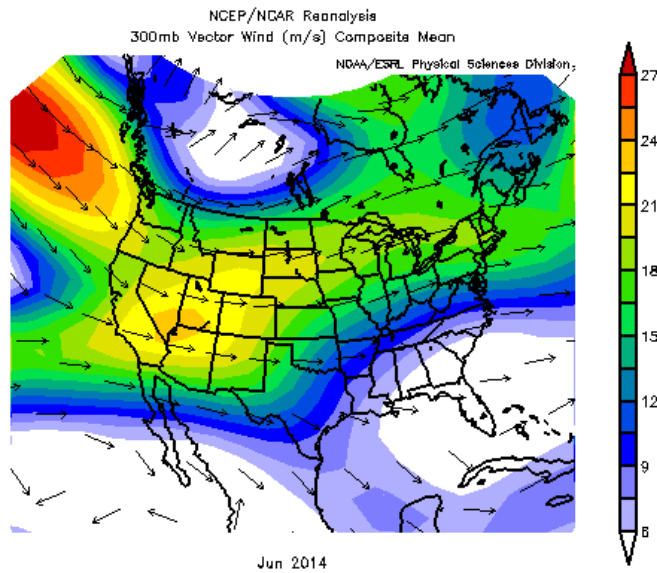

Figure 30.   Wind directions over North America at 30,000 ft., from [18]

Figures 29 and 30 show that the prevailing winds have a general westerly direction at 30,000 ft. and a negligible wind component at the surface. For the Monte Carlo simulation, a normal distribution was used with a 1-σ variation in wind magnitude of $wind_{mag} = 17 \frac{m}{s}$. Once the winds were broken down into components, they were multiplied by the Gaussian random number, n, with $\mu = 0$ and $\sigma = 1$ then multiplied by

the inverse of the transformation matrix given by (25). This was used to convert the wind vector to ECI to be used in the dynamics equations.

$$\begin{bmatrix} wind_N \\ wind_W \\ wind_U \end{bmatrix} = \begin{bmatrix} wind_{mag} * n \\ -wind_{mag} * n \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} wind_x \\ wind_y \\ wind_z \end{bmatrix} = C^T \begin{bmatrix} wind_N \\ wind_W \\ wind_U \end{bmatrix} \tag{31}$$

The *x, y,* and *z* components of the wind variation were the added to the relative wind velocity in the dynamics equations given by (32).

$$v_{relative} = v - \Omega \times r + wind \tag{32}$$

In (32), $\Omega \times r$ is the rotation velocity of the atmosphere based on Earth's angular rotation vector (33) and the current value of *r*.

$$\Omega = \begin{bmatrix} 0 \\ 0 \\ 7.2921158553e^{-5} \end{bmatrix} (rad / s) \tag{33}$$

Figures 31 through 32 are the resulting plots of the trajectories from the Monte Carlo simulation and the trajectory endpoints. Figure 31 is the full trajectories from the Monte Carlo while Figures 32 and 33 are the trajectory endpoints in north-up and north-west, respectively.
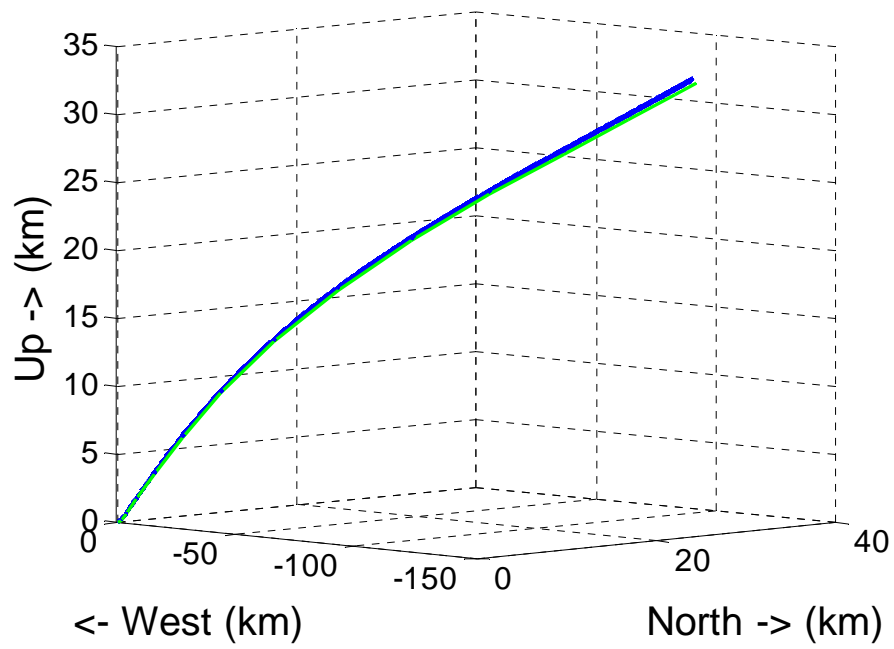
Figure 31.   Monte Carlo simulation for wind uncertainty
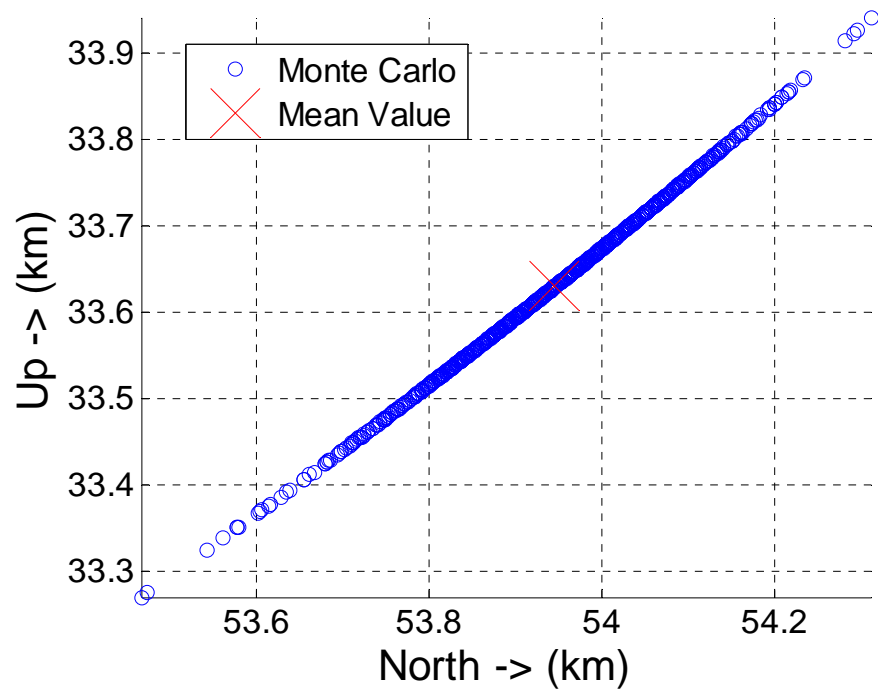

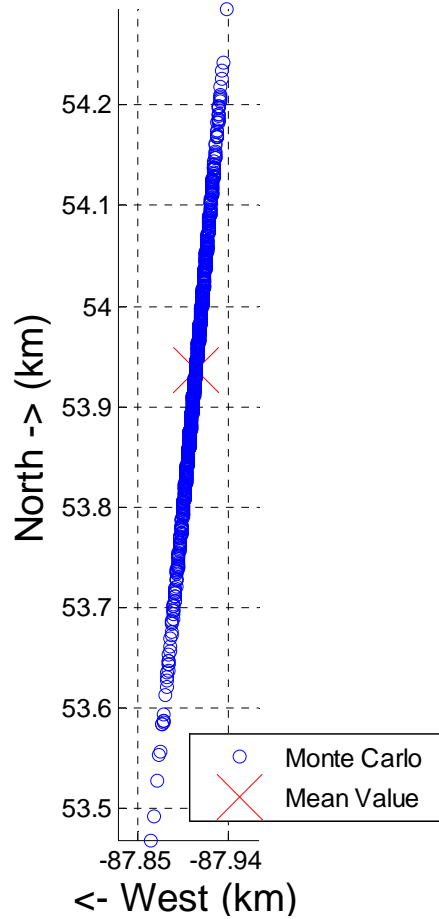Figure 32.   Plot of endpoints from Monte Carlo for wind in north-up

Figure 33.   Plot of endpoints from Monte Carlo for wind in north-west

The variance of the wind in the north-up plan is consistent with the variance in temperature in the same plane. Wind has a much smaller variance in the West direction which could suggest that the wind variation has a smaller effect on the launch trajectory than temperature.

E.   TEMPERATURE AND WIND VARIATION

The last part of the Monte Carlo simulation involved testing variations in both temperature and wind described by (30) and (32). Figures 34 through 36 are the trajectories that were obtained and the trajectory endpoints. Figure 34 is the full trajectory obtained from the Monte Carlo while Figures 35 and 36 are the trajectory endpoints in north-up and north-west, respectively.
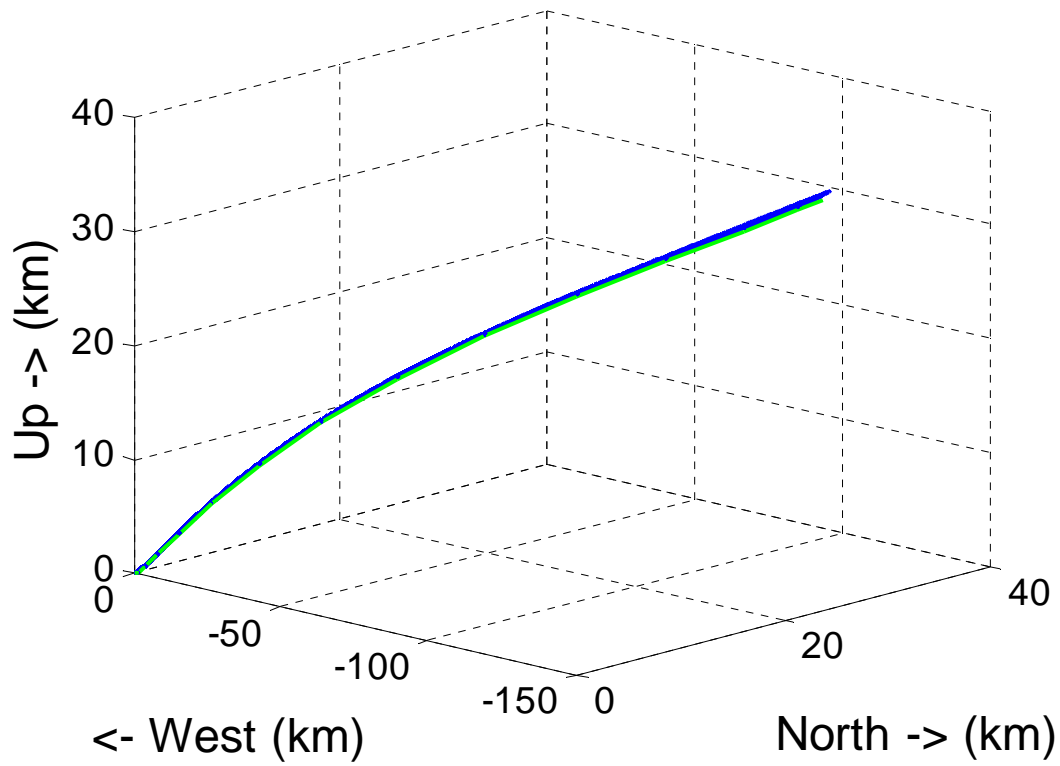
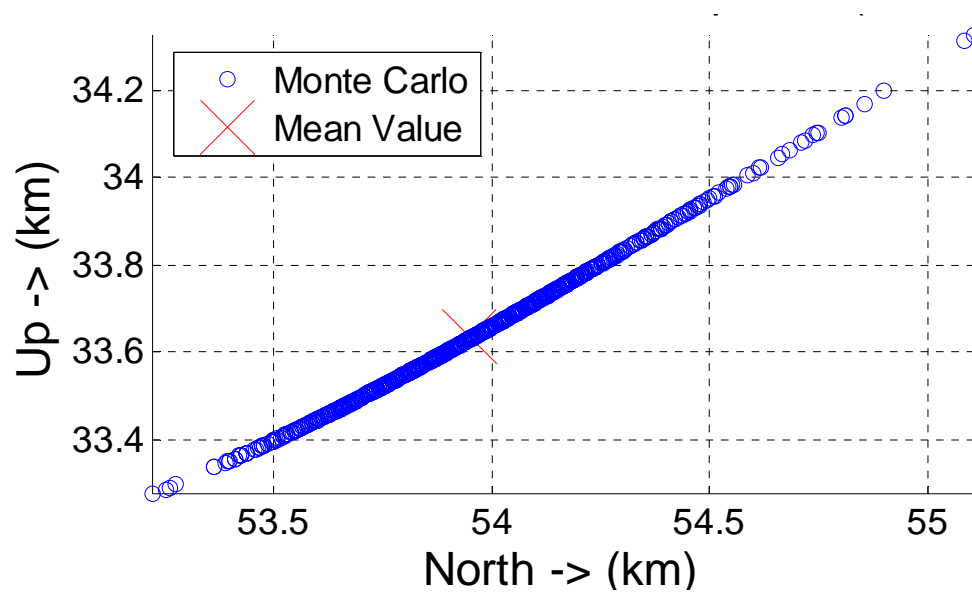Figure 34.   Monte Carlo simulation for temperature and wind uncertainty

Figure 35.   Plot of endpoints for Monte Carlo of temperature and wind in north-up
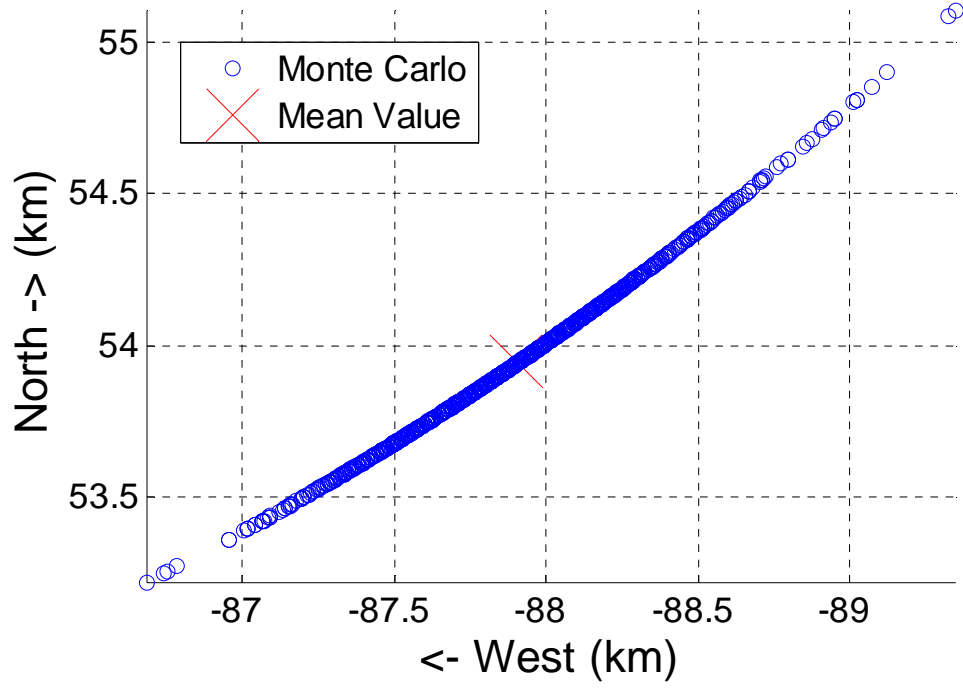
Figure 36.   Plot of endpoints for Monte Carlo of temperature and wind in north-west

The trajectories and trajectory endpoints are consistent with the two previous Monte Carlo simulations. Tables 3 and 4 show the statistical data from the three simulations. The standard deviation of the position endpoints and the mean final velocity are the two most important variables that need to be evaluated.

Simulation	μ_N (km)	μ_W (km)	μ_U (km)	σ_N (km)	σ_W (km)	σ_U (km)
Temp	53.934	-87.895	33.621	0.390	0.402	0.247
Wind	53.945	-87.908	33.630	0.123	0.011	0.098
Both	53.949	-87.898	33.635	0.269	0.390	0.150

Table 3.   Endpoint position means and standard deviations

Simulation	μ_{v_N} (km/s)	μ_{v_W} (km/s)	μ_{v_U} (km/s)	σ_{v_N} (km/s)	σ_{v_W} (km/s)	σ_{v_U} (km/s)
Temp	2.151	-3.710	1.116	0.015	0.020	0.011
Wind	2.151	-3.710	1.117	0.0035	$1.587e^{-4}$	0.002
Both	2.152	-3.710	1.117	0.012	0.020	0.008

Table 4.   Endpoint velocity means and standard deviations

From Table 3, it can be seen that neither of the uncertainties considered had a significant effect on the mean final position, but the temperature uncertainty gave the largest spread of endpoints. This suggests that more emphasis needs to be placed on predicting temperature than predicting wind patterns. In Table 4, neither of the uncertainties had any appreciable effect on the mean final velocity of the launch vehicle. Similar to before, the wind uncertainty had much smaller effect than temperature. This further confirms the conclusion that temperature has a larger influence on the launch trajectory than wind.

F.   SUMMARY

In this chapter, a model was developed to more accurately predict atmospheric density. This lead to the formulation of a single equation that can predict atmospheric density as a function of temperature offset and altitude. Next, Monte Carlo simulations were performed to assess the effects of uncertainties in the model. The uncertain variables chosen were temperature and wind based on the fact that these quantities have the potential to produce the greatest effects on the launch trajectory. The results from the simulations were plotted as trajectories and their respective endpoints. The statistical analysis showed that temperature had the largest effect on the launch trajectory of the three Monte Carlo simulations that were run. Based on the results from this chapter, the nominal solution may be sufficient for control of the first stage.

THIS PAGE INTENTIONALLY LEFT BLANK

VI.   CONCLUSIONS AND FUTURE WORK

A.   CONCLUSION

The stated goals for this thesis were achieved in that a rather convenient and simplistic model was created to expedite the process to create an initial launch trajectory for the first stage. The optimal trajectory that was created using DIDO was deemed to be feasible after a series of verification and validation tests. This was done with derived equations from Chapter III and propagation of the controls and plotting the results against the DIDO solution. Using the optimal trajectory, a series of uncertainties were placed on the simulation to analyze the sensitivity of the solution. The Monte Carlo simulations produced small deviations in the endpoint positions and had little effect on the vehicle's final velocity. From here, the nominal solution could be sufficient to control the starting point for the second stage, pending additional analysis. There is substantial room for future work that can be done in this area.

B.   HIGHER FIDELITY MODEL

When analyzing the dynamics portion of this model, the equations used were a simplification of reality. For example one of the assumptions in the model was that the thrust profile is constant over the period of the launch. Depending on the type of rocket, very different thrust profiles exist. This thesis does not address the design of rocket motors but more can be added to incorporate thrust profiles for various rocket motors used in industry. More emphasis should also be placed on the aerodynamics portion to produce more accurate approximation of aerodynamic drag. In the same fashion, models used for the Earth's gravitational force did not account for the oblateness of the earth or mass distribution. The atmospheric model that was used relies on a sixth order polynomial created as a curve fit base on observations from the 1976 standard atmospheric model that only goes until 30,000 ft. in increments of 1,000 ft. While the developed model seems to accurately estimate the atmospheric condition, there is still room for increasing the fidelity of the model. While most of the time higher order terms

are neglected, the combination of multiple "negligible" terms can have an appreciable effect.

C.   ADDITIONAL STAGES

Future work should take into account multiple stages to create a more realistic launch to orbit. The transition from single stage to multiple stages would be rather seamless. A separate function block would need to be written such that the starting position for the subsequent stages would be the ending points from the previous stages. In the same way the Monte Carlo simulation could be run to assess the sensitivity of the control for each stage.

D.   CODE ROBUSTNESS

Other launch trajectory generation tools should be compared against the results of this thesis. The time to formulate an optimal trajectory would be the most important metric to determine the effectiveness of this thesis research. Other areas might include how other trajectory generation tools perform their uncertainty analysis.

LIST OF REFERENCES

[1]   I. M. Ross and M. Karpenko, "A review of pseudospectral optimal control: From theory to flight," *Annual Reviews in Control,* vol. 36, no. 2, pp. 182–197, 2012.

[2]   DIDO (software) (n.d.). *Wikipedia.* [Online]. Available: http://en.wikipedia.org/wiki/DIDO_(software). Accessed Oct. 22, 2014.

[3]   C. Wilson, "Network centric warfare: Background and oversight issues for Congress," Congressional Research Service, Washington, DC, 2004.

[4]   B. S. Singh, "Shaping the land battle through remote sensing and satellite-imagery," *Strategic Analysis*, vol. 23, no. 11, p. 1909, 2000.

[5]   Launch Vehicles. (n.d.). NASA [Online]. Available: http://www.nasa.gov/externalflash/ISSRG/pdfs/launchvehicles.pdf. Accessed Oct. 22, 2014.

[6]   Advanced Space Transportation Program: Paving the Highway to Space. (n.d.). NASA. [Online]. Available: http://www.nasa.gov/centers/marshall/news/background/facts/astp.html_prt.htm. Accessed Oct. 22, 2014.

[7]   Program to Optimize Simulated Trajectories II. (2014, Jun. 10). NASA. [Online]. Available: http://post2.larc.nasa.gov/. Accessed Oct. 22, 2014.

[8]   D. Nelson, "Qualitative and Quantitative Assessment of Optimal Trajectories by Implicit Simulation (OTIS) and Program to Optimize Simulated Trajectories (POST)," 2001. [Online]. Available: http://www.ssdl.gatech.edu/papers/mastersProjects/NelsonD-8900.pdf. Accessed Oct. 22, 2014.

[9]   P. A. Ritter, "Optimization and Design for Heavy Lift Launch Vehicles," M.S. thesis, University of Tennessee, 2012. [Online]. Available: http://trace.tennessee.edu/utk_gradthes/1197. Accessed Oct. 22, 2014.

[10]   "Ch 3 Systems Approach to Guidance & Control," class notes for Spacecraft Guidance & Control, Dept. of Mechanical and Aerospace Engineering, Naval Postgraduate School, Monterey, CA, summer 2013.

[11]   "Ch 1 Review of AE3830," class notes for Astrodynamic Optimization, Dept. of Mechanical and Aerospace Engineering, Naval Postgraduate School, Monterey, CA, fall 2013.

53

[12]   "Ch 4 Constrained Dynamic Optimization," class notes for Astrodynamic Optimization, Dept. of Mechanical and Aerospace Engineering, Naval Postgraduate School, Monterey, CA, fall 2013.

[13]   D. D. Morrison, J. D. Riley and J. F. Zancanaro, "Multiple shooting method for two-point boundary value problems," *Communications of the ACM,* vol. 5, no. 12, 1962.

[14]   I. M. Ross, C. D'Souza, F. Fahroo and J. B. Ross, "A Fast Approach to Multi-Stage Launch Vehicle Trajectory Optimization," *AIAA Guidance, Navigation, and Control Conference Exhibit,* pp. 4–7, 2003.

[15]   Real-Time Space Shuttle in Google Earth (Beta Version). (n.d.). NASA. [Online]. Available: http://www.nasa.gov/mission_pages/shuttle/shuttlemissions/shuttle_google_earth.html. Accessed Oct. 10, 2014.

[16]   C. Z. Mooney, *Monte Carlo Simulation*, Thousand Oaks, CA, Sage, 1997, pp. 3–5.

[17]   1976 standard atmosphere calculator. (n.d.). Digital Dutch. [Online]. Available: http://www.digitaldutch.com/atmoscalc/. Accessed Oct. 22, 2014.

[18]   Monthly/Seasonal Climate Composites. (n.d.) NOAA, Earth Systems Research Laboratory, Physical Sciences Division. [Online]. Available: http://www.esrl.noaa.gov/psd/cgi-bin/data/composites/printpage.pl. Accessed Jul. 15, 2014.

INITIAL DISTRIBUTION LIST

1.	Defense Technical Information Center
	Ft. Belvoir, Virginia

2.	Dudley Knox Library
	Naval Postgraduate School
	Monterey, California